

***Ein Rahmenwerk für das Prozessdesign zur Identifikation,
Klassifikation und Umsetzung von Anforderungen***

-

Dargestellt an der Konzeption des Prozesskonfigurators

Von der Universität Bayreuth
zur Erlangung des Grades eines
Doktors der Naturwissenschaften (Dr. rer.nat.)
genehmigte Abhandlung

von

Dipl.-Wirtsch.-Ing. Stephanie Meerkamm

geboren in Krefeld-Uerdingen

1. Gutachter: Prof. Dr. Stefan Jablonski
2. Gutachter: Ass. Prof. Dr. Jan Recker

Tag der Einreichung: 07. Oktober 2011

Tag des Kolloquiums: 27. März 2012

Zusammenfassung

Prozessmanagement umfasst die Identifikation und Analyse der Unternehmensabläufe sowie deren Dokumentation und die Ausführung der Prozesse inklusive deren Steuerung. Dies sollte mit einer stetigen Verbesserung der Prozesse verbunden sein. Im Hinblick auf die konkrete Realisierung dieses Management-Ansatzes ist eine Fokussierung auf die Phase der Modellierung, welche die Identifikation sowie Dokumentation der Prozesse umfasst, zu beobachten. Dieser Phase geht das sog. *Prozessdesign* voraus. Auf Basis einer eingehenden Anforderungsanalyse erfolgt die Entwicklung und Evaluierung von Artefakten, wie zum Beispiel Konstrukte und/oder Methoden. Das vollständige Spektrum an Handlungsmöglichkeiten innerhalb des Prozessdesigns wird jedoch nur selten genutzt. Meist werden vorhandene (Standard-) Modellierungssprachen ausgewählt ohne diese in irgendeiner Art individuell zu gestalten. Dies wirkt sich oft negativ auf die Qualität der damit erstellten Prozessmodelle aus, aber auch auf die der Modellierung nachfolgenden Phasen des Management-Ansatzes wie Ausführung und Controlling.

Im ersten Teil der vorliegenden Arbeit wurde ein daher methodisches Rahmenwerk für das Prozessdesign entwickelt. Aufgrund der engen Kopplung an die nachfolgende Phase der Prozessmodellierung wird als Grundlage eine *Meta-Modell-Hierarchie* verwendet, die die Entwicklung, Anpassung sowie Definition von (Meta) Modellen vorsieht. Diese wurde explizit um eine *Designkomponente* inklusive der initialen Anforderungsanalyse erweitert. Damit liegt ein flexibles Vorgehensmodell für die Durchführung der Designphase vor, das dabei vor allem auch die Definition von Modellierungssprachen vorsieht. Das gesamte Rahmenwerk ist zudem nicht auf eine bestimmte Modellierungssprache oder Anwendungsdomäne der Prozesse ausgerichtet, sondern generisch konzipiert.

Im zweiten Teil der Arbeit wird ein Anwendungsfall für das zuvor entwickelte Rahmenwerk vorgestellt. Aus der beispielhaft identifizierten Menge an Anforderungen wurde die des Managements *variantenreicher Prozessmodelle* ausgewählt und das Konzept eines *Prozesskonfigurators* entwickelt. Im Hinblick auf die Verwendung der variantenreichen Prozessmodelle wurde ein *gestufter Konfigurationsprozess* entwickelt. Dieser leitet den Anwender auf der einen Seite durch den Konfigurationsprozess, auf der anderen Seite wird ihm ein Höchstmaß an Freiheit gewährt, wann er welche variantenbezogene Entscheidung treffen möchte. Für die Darstellung der Varianten in einem Modell wurde ein bereits existierendes Konzept zur Abbildung variantenreicher Strukturen in Form des sog. mereologischen Graphen verwendet, das an die Eigenschaften von Prozessen angepasst wurde. Damit kann die gewünschte kompakte sowie strukturierte Modellierung der Varianten in einem Modell realisiert werden. Dieses *Modellierungskonzept* konnte als Prototyp in einem entsprechenden Modellierungswerkzeug implementiert werden.

Abstract

Process management comprises the identification and analysis of procedures within an organization, their documentation and, finally, the execution of the processes inclusive of the monitoring and controlling which is supposed to lead to continuous improvement of these processes. With regard to the implementation of this management approach there is a focus on the phase of modeling which comprehends the identification as well as the documentation of the processes. The so called *process design* precedes this phase. On the basis of an exhaustive requirement analysis the development and evaluation of artifacts, for example constructs and/or methods, takes place. The complete scope of action of process design is, however, used quite rarely. Given (standard) process modeling languages are normally selected and no individual design work is carried out. This has a negative impact on the quality of the resulting process models and the phases which follow the modeling, such as, for example, execution and controlling.

In the first part of the work in hand a methodical *framework for the process design* has been developed. Due to the close conjunction with the subsequent modeling phase a *meta-model-hierarchy*, which allows for the development, adaption and the definition of (meta) models, is used as a basis. This has been extended explicitly by a *design component* which also includes the initial requirement analysis. By this means a flexible procedure model for the accomplishment of the design phase is available, which mainly envisages the definition of modeling languages. In addition, the framework is not focused on a particular modeling language or application domain, but is conceived generically.

In the second part of this work a user case study of the previously developed framework is presented. From the requirements which have been identified and exemplified, the management of variant-rich process models has been selected. For this, the concept of a *process configurator* has been developed and, in order to facilitate the usage of the variant-rich process models, a *staged configuration process* has been developed. This, On the one hand, guides the user during the configuration process while, on the other hand, the user is accorded the maximum freedom of action concerning the point of time at which he wants to take the variant related decision. In order to model the variant-rich processes efficiently an already existing concept for illustrating variant-rich structures, the so called *mereological graph*, has been adapted according the characteristics of processes. With the adoption of this concept the desired compact and structured method of modeling variants in one single model could be achieved. It was finally possible to implement the modeling concept as a first prototype in an appropriate modeling tool.

Vorwort

Auf dem Weg zu der vorliegenden Arbeit haben mich zahlreiche Leute begleitet und zu ihrem erfolgreichen Abschluss beigetragen. Ich möchte mich an dieser Stelle für ihre Unterstützung, ihre Ratschläge und ihre aufgebrauchte Zeit bedanken.

Mein Dank gilt Herrn Prof. Dr.-Ing. Stefan Jablonski, an dessen Lehrstuhl diese Arbeit entstanden ist. Herrn Ass. Prof. Dr. Jan Recker danke ich für die Übernahme des Zweitgutachtens.

Ich möchte mich bei meinem Kollegen Bernhard Volz bedanken, der mich mit viele und langen Gesprächen unterstützt hat und meine Ingenieur-Fragen zur Informatik stets mit großer Geduld beantwortet hat.

Christine Leinberger, der gute Seele des Lehrstuhls, danke ich für selbstlose Unterstützung während meiner Zeit am Lehrstuhl. Mit ihrer fröhlichen und optimistischen Art, ihrem beispielhaft vorgelebten Durchhaltevermögen hat sie mich auf ihre Art und Weise bei meiner Promotion unterstützt. Bernd Schlesier danke ich für die zahlreichen, herzlichen Gespräche und seine stetige Hilfsbereitschaft.

Wenn ich ihn nicht kennengelernt hätte, hätte ich wahrscheinlich niemals mit einer Promotion begonnen: Sascha Müller-Feuerstein. Er hat mich mit viel Fingerspitzengefühl, Humor (auch wenn er bei meinen Lachanfällen manchmal seine Zweifel hatte - ich habe ihn ernst genommen) und ungemeinem Fachwissen in die Welt der Wissenschaft bzw. in die Welt der Informatik eingeführt. Für die vielen langen Gespräche, in denen er stets großes Interesse an meiner fachlichen wie auch persönlichen Weiterentwicklung hatte, danke ich ihm ganz herzlich. Ich hoffe ihm mit dieser Arbeit ein wenig von dem zurückgeben zu können, was er mir gegeben hat.

Danken möchte ich dem SC Uttenreuth. Das Leben im Verein hat mich persönlich geprägt und Werte vermittelt, die ich hoffentlich noch vielen Menschen innerhalb und außerhalb des Vereins weitergeben kann. Die Zeit, die ich dort verbracht habe, möchte ich nicht missen.

Dass ich in Bayreuth nicht komplett in der Dissertation versunken bin, habe ich verschiedenen Leuten zu verdanken. Dazu beigetragen hat die Tennisabteilung des SV Heinersreuth (Ich habe dort das bisher beste Tennis meiner „Karriere“ gespielt). Zusammen mit den „Siemensianern“ aus Kemnath durfte ich, als „Externe“, viele gemütliche Abende verbringen. Mit der „Spielegruppe“ habe ich einen Kreis von Menschen gefunden, zu denen ich hoffentlich noch lange Kontakt halten kann, auch wenn ich nun nicht mehr direkt vor Ort bin. Last but not least möchte ich ganz besonders Sabrina Uhrig bedanken. Direkt am Nachbarlehrstuhl in der Angewandten Informatik haben wir uns gegenseitig bei unseren Arbeiten unterstützt; sie ist für mich mehr als eine Kollegin geworden.

Ein Dankeschön geht an meinen Studienfreundeskreis, der „MB-Crew“ aus Erlangen für die Treue, trotz der wenigen Zeit, die ich oftmals für sie aufgebracht habe. Ganz besonders danke ich Stefanie Klingner und Stephan Schuh. Roy Remannifield danke ich für die Reviews meines Papers.

Zu aller Letzt, jedoch am allermeisten, möchte ich mich bei denen bedanken, die mich auf dem gesamten Lebensweg bis hier hin begleitet haben. Die notwendige Kraft und das Durchhaltevermögen, um die Promotion letztendlich erfolgreich abschließen zu können, haben mir vor allem der Rückhalt durch meine Eltern und meine Zwillingschwester ermöglicht. Sie haben mich unermüdlich unterstützt, immer wieder aufgebaut und stets an mich und meine Fähigkeiten geglaubt. Der Wert einer Familie lässt sich nur schwer in Worten fassen, ich sage einfach von ganzen Herzen: Danke!

Inhaltsverzeichnis

Zusammenfassung	I
Abstract	III
Vorwort	V
Inhaltsverzeichnis	IX
Abbildungsverzeichnis	XIII
Tabellenverzeichnis	XV
Kapitel 1 Einführung	1
1.1 Motivation	1
1.1.1 Modellierung und der Aspekt des Designs	1
1.1.2 Prozessdesign	4
1.1.3 Konfigurierbare Prozessmodelle	5
1.2 Problemstellungen und Zielsetzungen	8
1.2.1 Prozessdesign	8
1.2.2 Konfigurierbare Prozessmodelle	9
1.3 Publikationen	11
1.4 Aufbau der Arbeit	12
Kapitel 2 Prozessmanagement – Stand in Forschung und Praxis	15
2.1 Der Prozessbegriff	15
2.2 Anwendungsbereiche des Prozessbegriffs	17
2.2.1 Produktentwicklungsprozess	18
2.2.2 Medizinischer Prozess	20
2.2.3 Verwaltungstechnischer Prozess an der Universität	22
2.2.4 Fazit	24
2.3 Prozessmanagement	24
2.3.1 Prozessmanagement – Stand in der Forschung	25
2.3.2 Prozessmanagement – Stand in der Praxis	29
2.3.3 Fazit bzgl. dem Stand in Forschung und Praxis	47
Kapitel 3 Design und Modellierung – Definition und Grundlagen	51
3.1 Design und Modellierung – eine Abgrenzung	51
3.1.1 Design	51
3.1.2 Modellierung	55
3.1.3 Fazit	58
3.2 Methodische Grundlagen des Designs	59
3.2.1 Pragmatisches Prozessdesign	60
3.2.2 Theoretisches Prozessdesign	63

3.2.3	Fazit zu den Grundlagen des Designs	68
3.3	Methodische Grundlagen der Modellierung	70
3.3.1	Modellierungssprachen	70
3.3.2	Standard- vs. Anwendungsbezogene Modellierungssprachen.....	81
3.3.3	Meta-Modellierung	86
3.3.4	Fazit zu den Grundlagen der Modellierung	89
3.4	Fazit	90
Kapitel 4	Design und Modellierung – Anforderungen	95
4.1	Übersicht über die Projekte	95
4.2	Anforderungen.....	97
4.2.1	Phasen	97
4.2.2	Kreative bzw. „unscharfe“ Prozesse	100
4.2.3	Umfangreiche Prozessmodelle.....	102
4.2.4	Zeitlicher Aspekt.....	106
4.2.5	Komplexe Entscheidungssituation	108
4.2.6	IT-relevante Prozessabschnitte.....	110
4.2.7	Sprünge	112
4.2.8	Unterschiedliche Sichtweisen	115
4.2.9	Unterschiedliche Datentypen	117
4.2.10	Unterschiedliche Prozesstypen	119
4.3	Fazit – Handlungsbedarf.....	120
Kapitel 5	Design und Modellierung – im Rahmen der Meta Modell Hierarchie	121
5.1	Traditionelle Betrachtung von Design und Modellierung.....	121
5.2	Erweiterte Interpretation von Design und Modellierung.....	123
5.3	Klassifikation der Anforderungen	125
5.4	Das Vorgehensmodell	127
5.5	Erläuterungen der einzelnen Ebenen	130
5.5.1	M3 – Design und Modellierung	130
5.5.2	M2 – Design und Modellierung	131
5.5.3	M1 – Design und Modellierung	133
5.5.4	M0 – Design und Modellierung	135
5.5.5	M0 – Ausführung und Monitoring/ Controlling	136
5.6	Prototypische Realisierung in OMME	137
5.6.1	Konzept und Architektur von OMME	137
5.6.2	Use Case.....	138
5.7	Zusammenfassung	139
Kapitel 6	Der Prozesskonfigurator	143
6.1	Motivation	143
6.2	Der Begriff der Variante.....	144
6.2.1	Definition Variante	145
6.2.2	Abgrenzung Variante - Entscheidungspunkte.....	147
6.2.3	Abgrenzung Variante - Version	151
6.3	Ansätze zur Modellierung von Varianten.....	152
6.4	Ansätze des prozessorientierten Variantenmanagements.....	154
6.4.1	C-iEPC	154
6.4.2	PESOA	158
6.4.3	Provop	159

6.4.4	Fazit.....	162
6.5	Der Prozesskonfigurator	162
6.5.1	Anforderungen an den Prozesskonfigurator	163
6.5.2	Das Konzept des Prozesskonfigurators	166
6.5.3	Anforderungen an die konfigurierbare Datenbasis	173
6.5.4	Das Konzept der konfigurierbaren Datenbasis	175
6.5.5	Das Generische Datenmodell.....	177
6.5.6	Das Anwenderorientierte Prozessmodell	189
6.5.7	Evaluation des Konfigurationskonzeptes.....	193
6.5.8	Prototypische Implementierung der Datenbasis in OMME.....	200
6.6	Fazit	207
Kapitel 7	Zusammenfassung	209
7.1	Resumée	209
7.2	Ausblick.....	212
Literaturverzeichnis		213
Anhang A	Leitfaden - Interview	241
Anhang B	BPMN Events	233

Abbildungsverzeichnis

Abbildung 1-1 Modelle aus der Produktentwicklung [BuAn89]	2
Abbildung 1-2 Teilphasen des Designs.....	3
Abbildung 1-3 Varianten und ein integriertes Prozessmodell	6
Abbildung 1-4 Schematischer Aufbau der Arbeit.....	13
Abbildung 2-1 Beispiel – Produktentwicklungsprozess aus [Meer07b].....	20
Abbildung 2-2 Beispiel – Medizinischer Prozess [eigene Quellen]	21
Abbildung 2-3 Beispiel – Universitätsbezogener Prozess [UnBa10]	23
Abbildung 2-4 Prozesslebenszyklus.....	25
Abbildung 3-1 Beispiel zeichenhaft-semantische Funktion.....	52
Abbildung 3-2 Bildung kompositer Schritte aus [ReLi04]	61
Abbildung 3-3 Resequencing aus [ReLi04]	61
Abbildung 3-4 Produktdatenmodell aus [ReNV07].....	65
Abbildung 3-5 Beispiel UML-Aktivitätsdiagramm	71
Abbildung 3-6 Beispiel EPK-Modell.....	74
Abbildung 3-7 Beispiel BPMN-Modell	76
Abbildung 3-8 Beispiel AOPM-Modell.....	79
Abbildung 3-9 Meta-Modell-Hierarchie zu AOPM [JaVD09]	86
Abbildung 3-10 Beispiele Prozessmodellierungssprachen aus [Reck08]	90
Abbildung 4-1 Phasen mit AOPM im i>pm modelliert	99
Abbildung 4-2 Meilenstein mit AOPM im i>pm modelliert.....	100
Abbildung 4-3 Beispiel komposite Modellierung.....	105
Abbildung 4-4 Komplexe Entscheidung mittels elementaren Entscheidern (AOPM) aus [Meil05].....	110
Abbildung 4-5 Evidenzbasierter Entscheider aus [Meil05]	110
Abbildung 4-6 Sprünge im Prozessmodell.....	112
Abbildung 4-7 Darstellung eines Prozesses mit Swimlanes	117
Abbildung 5-1 Traditionelle Einordnung von Design und Modellierung.....	121
Abbildung 5-2 Erweiterte Interpretation von Design und Modellierung	123
Abbildung 5-3 Klassifikation von Anforderungen gem. der Meta-Modell-Hierarchie	126
Abbildung 5-4 Vorgehensmodell für Design und Modellierung	128
Abbildung 5-5 Beispielhafte Interpretation von "node & edge" (M3) auf der Ebene M2.....	132
Abbildung 5-6 Screenshot OMME [eigene Quellen].....	138
Abbildung 5-7 Deklaration eines Knoten (= Node) im LLM - M3	139
Abbildung 5-8 Von der Revolution zur Evolution.....	141
Abbildung 6-1 Entscheidungspunkt mit Alternativen.....	148
Abbildung 6-2 Varianten mit variation points und variant options	149
Abbildung 6-3 Variante 1	150
Abbildung 6-4 Variante 2.....	150
Abbildung 6-5 Versionen – Varianten	151
Abbildung 6-6 C-iEPC aus [LDHM11]	156

Abbildung 6-7 Nach PESOA annotiertes BPMN-Prozessmodell.....	158
Abbildung 6-8 Beispiel mit Aufsetzpunkten aus [HaBR08a].....	160
Abbildung 6-9 Kategorisierung der Anforderungen	163
Abbildung 6-10 Konzept des Prozesskonfigurators.....	166
Abbildung 6-11 ER-Diagramm.....	167
Abbildung 6-12 Konzeptueller Aufbau der Datenbasis	176
Abbildung 6-13 Generisches Datenmodell [eigene Quelle]	179
Abbildung 6-14 Implikation zwischen verschiedenen Aspekten.....	181
Abbildung 6-15 variation point und variant options	182
Abbildung 6-16 Hierarchisch angeordnete variation points [eigene Quellen].....	183
Abbildung 6-17 Darstellung der konfigurierbaren Stellen.....	186
Abbildung 6-18 Konfigurierbares AOPM-Prozessmodell.....	189
Abbildung 6-19 Rechnung schreiben – integriertes AOPM-Prozessmodell.....	194
Abbildung 6-20 Rechnung schreiben - Variante 1	195
Abbildung 6-21 Rechnung schreiben – Variante 2	195
Abbildung 6-22 Rechnung schreiben – generisches Datenmodell.....	197
Abbildung 6-23 Screenhoot OMME	201
Abbildung 6-24 Deklaration des Graphen auf M2.....	201
Abbildung 6-25 Deklaration der Knoten auf M2	202
Abbildung 6-26 Deklaration der Kanten auf M2	203
Abbildung 6-27 Deklaration der Verbindungen auf M2.....	203
Abbildung 6-28 Screenshot eines generisches Datenmodells in OMME.....	204
Abbildung 6-29 Deklaration des variation points auf M2.....	205
Abbildung 6-30 Deklaration der organisatorischen variant option auf M3	205
Abbildung 6-31 Konfigurierbares AOPM-Modell in OMME	206
Abbildung 7-1 Konkretisierte Teilphasen des Designs.....	210

Tabellenverzeichnis

Tabelle 2-1 Übersicht über die befragten Unternehmen	36
Tabelle 3-1 Beispiele Informationsobjekte EPK.....	53
Tabelle 3-2 Beispiele für Verwendung von Modellen	56
Tabelle 3-3 Design-Artefakte nach [HMPR04]	59
Tabelle 3-4 Gegenüberstellung pragmatisches und theoretisches Prozessdesign	69
Tabelle 4-1 Beispiele Phasen	97
Tabelle 4-2 Beispiele kreativer bzw. unscharfer Prozesse	100
Tabelle 4-3 Beispiele umfangreicher Prozesse	103
Tabelle 4-4 Beispiele zeitliche Perspektive	107
Tabelle 4-5 Beispiele komplexer Entscheidungssituationen.....	109
Tabelle 4-6 Beispiele ausführbare Prozessabschnitte	111
Tabelle 4-7 Beispiele Sprünge	113
Tabelle 4-8 Beispiele unterschiedlicher Sichtweisen.....	116
Tabelle 4-9 Beispiele Datentypen	118
Tabelle 4-10 Beispiele Prozesstypen.....	119
Tabelle 6-1 Prozessvorauswahl – Auswahl der variant options.....	184
Tabelle 6-2 Prozessausführung - Entscheidungsmöglichkeit I	185
Tabelle 6-3 Prozessausführung - Entscheidungsmöglichkeit II.....	185
Tabelle 6-4 Zuordnung der Elemente - Sprachunabhängigkeit	188
Tabelle 6-5 Abbildung generisches Datenmodell - AOPM	192

Kapitel 1 Einführung

1.1 Motivation

Während schon gegen Ende des letzten Jahrtausends Davenport und Short [DaSh90] sowie Hammer und Champy [HaCh93] die Bedeutung der Prozesse im Rahmen einer (Unternehmens-) Organisation herausgearbeitet haben, ist das *Management der Anwendungsprozesse* [JaBS99] heutzutage mehr denn je eine herausfordernde Aufgabe [GaGr09] [Hamm01] [SmFi03]. Die Globalisierung, die wirtschaftlichen Turbulenzen wie sie die Finanzkrise 2010 mit sich gebracht hat, gestiegene Compliance-Anforderungen an die Unternehmen, wie zum Beispiel im Zusammenhang mit dem Sarbanes-Oxley Act aus dem Jahr 2002, oder auch der technologische Fortschritt der eingesetzten Informationstechnologien erfordern eine entsprechende Organisation der Unternehmen und ihrer Abläufe.

Prozessmanagement umfasst die Identifikation und Analyse der Abläufe einer Organisation sowie deren Dokumentation, bis hin zur Ausführung der Prozesse samt ihrer Steuerung, was mit einer stetigen Verbesserung der Prozesse verbunden sein sollte [GaSc95] [ScSe08] [vBRo10]. Dabei geht es sowohl um organisatorische Aspekte im Hinblick auf die eingesetzten Ressourcen, als auch um die Unterstützung durch geeignete Informationstechnologien. Die genannten Aufgaben werden in Form eines sogenannten, sich wiederholenden, Prozesslebenszyklus organisiert [BeKR05] [zMRo04] (in Kapitel 2.3.1 folgt eine umfassend Erläuterung).

1.1.1 Modellierung und der Aspekt des Designs

Im Hinblick auf die konkrete Realisierung des Prozessmanagement-Ansatzes konnte in den letzten Jahren eine zunehmende Fokussierung auf die Identifikation sowie Dokumentation der (Geschäfts-) Prozesse beobachtet werden [DGRI06] [Rose06a]. Letzteres, die *Modellierung* der Prozesse, beinhaltet die graphische Abbildung der Abläufe inklusive der Angabe von Ressourcen (zum Beispiel Daten, Rollen, Werkzeuge [Agui04] [CuKO92] [JaBu96]). Dabei kommen Prozessmodellierungswerkzeuge mit einer zumeist graphischen Modellierungssprache bzw. -notation wie zum Beispiel BPMN [OMG11a] zum Einsatz. Prozessmodelle werden in den verschiedensten Anwendungsdomänen (unter anderem Medizin, Produktentwicklung, Verwaltung...) verwendet; ebenso vielfältig sind die Einsatzzwecke der Modelle (Training, Analyse, Simulation, Ausführung, Entwicklung von Informationssystemen zur automatisierten Prozessausführung, etc., [CuKO92] [DuAH05]).

[DevA04]), so dass sich die unterschiedlichsten Anforderungen an die Modellierungssprache und das Werkzeug ergeben.

Mit der Veröffentlichung des Artikels von Hevner [HMPR04] hat das Interesse bzgl. des sog. *Designs* im Bereich der Informationssysteme erheblich zugenommen. Zuvor sind dazu bereits zahlreiche Beiträge veröffentlicht worden wie zum Beispiel von [Glas99], [MaMG02], [MaSm95], [WaWE92] oder [Wino97]. Die Designphase umfasst demnach die Entwicklung und Evaluierung von Artefakten, wie zum Beispiel Methoden, Konstrukte oder Modelle, auf Basis konkreter Anforderungen. Im Kontext des Prozessmanagements handelt es sich um *Anforderungen* bzgl. der Gestaltung der Prozessmodelle und deren Ausführung. Diese Anforderungen werden umgesetzt in der Definition von Modellierungssprachen mit den entsprechenden Konstrukten [DuAH05]. *Die im Rahmen der Modellierung erstellten Prozessmodelle dienen der Designphase letztendlich als Instrument bzw. Hilfsmittel, um die Anforderungen und ihre Umsetzungskonzepte festzuhalten und diese dann weiterzuverarbeiten* [DaSh90]. Dies bedeutet, dass die Anforderungen zunächst in einer Modellierungssprache soweit wie möglich abgebildet werden und im weiteren Verlauf durch die Modellierung (oder dann durch die Ausführung) umgesetzt werden.

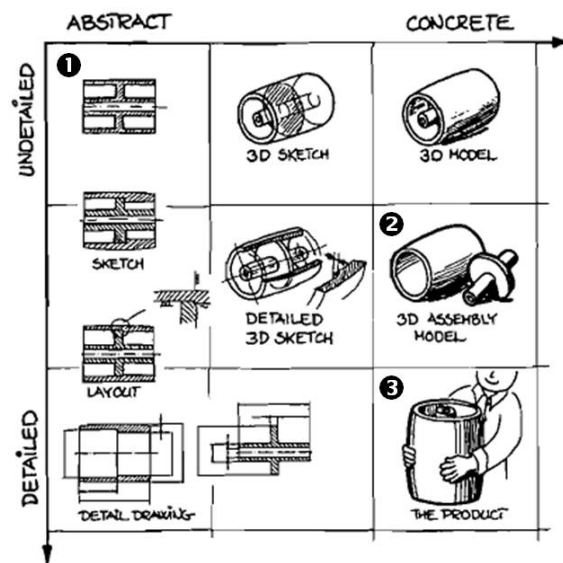


Abbildung 1-1 Modelle aus der Produktentwicklung [BuAn89]

Auch in der Domäne der Ingenieurwissenschaften haben das Design und die Erstellung von Modellen einen hohen Stellenwert [Dym94] [PaBe97]. Der Vorgang der Modellierung wird als „The Language of the Designer“ [Andr94] bzw. des Produktentwicklers angesehen - als Mittel zur Kommunikation und Dokumentation der Anforderungen und Analyseergebnisse. Wie bei [DaSh90] wird das Modell als Instrument der Designphase betrachtet; die in der Designphase erzielten Ergebnisse werden als Input für die Modellerstellung verwendet. Wie

unterschiedlich die Modelle je nach Einsatzzweck bereits innerhalb einer Domäne ausfallen können, zeigt Abbildung 1-1. In allen Feldern der Abbildung 1-1 wird prinzipiell dasselbe Objekt modelliert, jedoch mit unterschiedlichem Abstraktions- bzw. Detaillierungsgrad (= *Eigenschaften des Modells*). Die Eigenschaften stehen in engen Zusammenhang mit dem *Zweck (der Verwendung) des Modells*: so liefert ❶ in Abbildung 1-1 Informationen über das „Innenleben“ des Objektes; ❷ stellt dar, wie das Objekt zusammenzubauen ist, bei ❸ handelt es sich um ein physisch fassbares Objekt. Oftmals ergeben sich bei mehreren Anforderungen Widersprüche, die es aufzulösen gilt. So soll die Reibung des Wälzkörpers gering sein, zugleich soll jedoch ein bestimmter Festigkeitswert des Materials erfüllt werden.

So unterschiedlich die inhaltlichen Eigenschaften und Zwecke sind, so unterschiedlich sind auch die Arbeitsmittel, das heißt der Input, zur Modellerstellung: Für die Modellierung von ❶ sind im einfachsten Fall Bleistift und Papier notwendig, die für die Zeichnungserstellung notwendigen Normen und die geometrischen Angaben des Objekts. Bei ❸ muss ein entsprechender Werkstoff (zum Beispiel Stahl), das Herstellungswerkzeug (zum Beispiel ein Schleifgerät) inklusive einer geeigneten Verfahrensweise bzw. Methodik (zum Beispiel Nassschleifen mit Kühlmittelzugabe) vorhanden sein. Im Hinblick auf die Erstellung von Prozessmodellen muss gemäß den Charakteristiken der vorliegenden Prozesse eine Modellierungssprache mit den entsprechenden Modellierungselementen (zum Beispiel Prozesse, Daten, Rollen) spezifiziert werden.

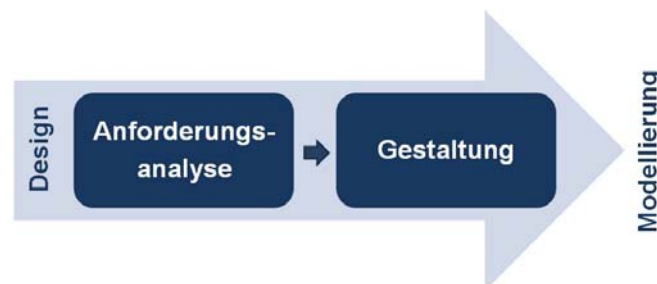


Abbildung 1-2 Teilphasen des Designs

Die genannten Anwendungsfälle können folgendermaßen verallgemeinert werden: im Rahmen des **Designs** kann zwischen der **Analyse- und der Gestaltungsphase** unterschieden werden (siehe Abbildung 1-2). Es müssen zunächst in der Anforderungsanalyse die *Eigenschaften des zu modellierenden Objektes* und dessen weiteren *Verwendung im Rahmen des dazugehörigen Lebenszyklus* definiert werden. Die Anforderungen gilt es nachfolgend in der Gestaltungsphase auf folgende Aspekte abzubilden: *Input bzgl. Informationen/Material, Werkzeug* sowie die dazugehörige *Verfahrensweise bzw. Methodik*. Diese Ergebnisse fließen dann als Input in die Modellierung ein.

Mit einer derart strukturierten Designphase wird die Grundlage für eine anforderungsgerechte Modellierung gelegt. So kann nach eingehender Anforderungsanalyse die entsprechende

Vorgabe eines Schleifverfahrens im Rahmen der Gestaltungsphase des Designs (Abbildung 1-2) bei ❸ verhindern, dass das als Wälzkörper in einem Lager verwendet Objekt nach der Herstellung zu übermäßigen Verschleiß führt. Dieser tritt auf bei zu hoher Reibung aufgrund nicht ausreichenden Feinschliffs des Bauteils; das Lager erhitzt sich und schlimmstenfalls erleidet die gesamte Maschine einen Schaden, was diverse weitere negative Folgen (zum Beispiel wirtschaftliche Folgen durch Produktionsausfall) mit sich bringt. Werden die Anforderungen erst später ermittelt, zum Beispiel wenn der Herstellungsprozess des zu schleifenden Bauteils bereits begonnen hat, können sie meist nur noch mit hohen Kosten und viel Aufwand eingebracht werden [RIRG09] [Rose06a]. Die einzelnen Aspekte, ihre Beziehungen und Wechselwirkungen zu einander, sind dann aber bereits festgelegt und können nur schwer bzw. nur mit weitreichenden Folgen geändert werden. So kann die geometrische Form nach dem Schleifvorgang lediglich verkleinert, nicht jedoch vergrößert werden; Schnittkanten müssen gegebenenfalls nachbearbeitet werden.

Das Design mit seiner Analyse- und Gestaltungsphase stellt ein sehr umfassendes und für die nachfolgenden Phasen einflussreiches Aufgabengebiet dar. Es ist daher nachvollziehbar, dass das Design in keiner Weise vernachlässigt werden darf, wenn qualitativ hochwertige Modellierungsergebnisse erzielt werden sollen, was sich wiederum positiv auf die Qualität ihrer Verwendung auswirkt. Die Beispiele aus der Produktentwicklung, speziell das der Fertigung, stehen für den erfolgreichen Einsatz des Designs.

1.1.2 Prozessdesign

Beim *Prozessdesign* im Rahmen des Prozessmanagements steht die Gestaltung von Input, Werkzeugen und Methodik für die Erstellung anforderungsgerechter Prozessmodelle im Vordergrund. Dazu müssen in der Anforderungsanalyse alle wesentlichen *Eigenschaften eines Prozessmodells* unter Berücksichtigung des jeweiligen Kontextes identifiziert werden. Ein Prozessmodell für ein Produktionsunternehmen muss zum Beispiel die Kennzahl Rüstzeit, die eine unproduktive Zeit darstellt, bei den für das Einrichten der Produktionsmaschinen notwendigen Schritte abbilden; die damit bei der Ausführung der Schritte erfassten Zeiten werden zum Ableiten von Qualitätsmanagementmaßnahmen verwendet [ScSe08]. Im medizinischen Bereich gilt es zur Unterstützung des medizinischen Personals komplexe Entscheidungssituationen während einer Behandlung mit allen Entscheidungskriterien und Entscheidungsfolgen verständlich abzubilden und gegebenenfalls informationstechnisch durch ein Entscheidungsunterstützungssystem zu unterstützen [Meil05]. An den Beispielen ist zu erkennen, dass auch der *Einsatzzweck von Prozessmodellen* identifiziert werden muss. Angefangen von der reinen Dokumentation in unterschiedlichen Detaillierungsgraden kann sich dieser bis hin zur automatisierten Ausführung der Prozessmodelle erstrecken.

Im Anschluss an die Anforderungsanalyse müssen im Bezug auf die Gestaltungsaspekte Input, Werkzeug und Methodik konkret die zu modellierenden *Inhalte* (zum Beispiel Prozessschritte mit der Angabe von In- und Output-Daten), eine adäquate

Modellierungssprache sowie das *Werkzeug* (zum Beispiel BPMN mit Aris Express [Soft11]) definiert werden.

Doch obwohl es inzwischen eine Vielzahl von Modellierungssprachen und Werkzeugen gibt [Sinu04], haben die Erfahrungen aus den Projekten des Lehrstuhls für Angewandte Informatik IV an der Universität Bayreuth in unterschiedlichen Anwendungsbereichen (u.a. Medizin [FMSJ09], Verwaltung bzw. Universität [UnBa11], Produktentwicklung [MePa08]) gezeigt, dass gerade die Definition der Modellierungssprache (und damit auch des Modellierungswerkzeuges) oftmals aus Zeit- und Kostengründen oder aus Bequemlichkeit vernachlässigt wird. So wurden in den Projekten stets dieselbe Modellierungssprache (und auch dasselbe Werkzeug) verwendet. Während für die Produktentwicklung und die Medizin zwar geringfügig spezielle Anpassungen der Modellierungssprachen vorgenommen wurden, war dies bei dem verwaltungstechnischen Projekt an der Universität nicht der Fall. Eine zusätzlich vom Lehrstuhl für Angewandte Informatik IV durchgeführte Interviewreihe an ausgewählten deutschen Unternehmen [Meer09] ergab, dass es bzgl. der Wahl einer Modellierungssprache und Werkzeug meist an einer kritischen Gegenüberstellung mit den auf dem Markt vorhandenen Produkten fehlt [MaSc97]. Die Entscheidung fällt allzu häufig auf die Auswahl einer Standardmodellierungssprache, wie zum Beispiel UML [OMG11e] oder BPMN [OMG11a], oder bereits vorhandenen anwendungsbezogene Modellierungssprachen. Diese sollen keinesfalls als ungenügend verurteilt werden. Doch kann davon ausgegangen werden, dass sie nicht alle Aspekte eines individuellen Prozesses anforderungsgerecht abbilden können [BICS07] [Rose06a]. Dies wirkt sich auch auf die nachfolgenden Phasen der Modellierung aber auch der Ausführung sowie des Monitorings und Controllings aus [RIRG09] [Rose06a], wie an dem Beispiel der Rüstzeit zu sehen ist. Wird dieses Prozessmodell zudem nicht in einer maschinenlesbaren Form erstellt, können die Angaben auch nicht als Grundlage zur Steuerung der Produktionsmaschine verwendet werden. Dieses Beispiel zeigt, dass die Individualität einer Anforderung die Entwicklung einer anwendungsbezogenen Prozessmodellierungssprache erfordert. Diese Vorgehensweise wird in Abbildung 1-2 veranschaulicht.

1.1.3 Konfigurierbare Prozessmodelle

Ein in den letzten Jahren viel diskutierte Design-Aspekte sind Varianten bzw. konfigurierbare Prozessmodelle [RovA07]. Der Modellierer steht nicht selten vor dem Problem, dass es zu ein- und demselben Prozess verschiedene, aber dennoch ähnliche Ausprägungen, sog. *Varianten* [HaBR08a], gibt. Auch in den am Lehrstuhl für Angewandte Informatik IV erstellten Prozessmodellen (mehr dazu in Kapitel 2.2 bzw. 4.1) konnten zahlreiche Varianten identifiziert werden. Um den verschiedensten Anforderungen des Wettbewerbsumfeldes und der Stakeholder je nach Branche, Organisation/Abteilung oder Projekt, etc. gerecht zu werden ist oftmals eine Vielzahl von ähnlichen, jedoch im Bezug auf einzelne Aspekte unterschiedlichen Prozessen zu definieren. So unterscheiden sich zum Beispiel die

Aufnahmeprozesse in einer Klinik, je nachdem ob der Untersuchungsplan von einer Krankenschwester oder aber von einem Arzt erstellt wurde. Da Varianten in vielen verschiedenen Anwendungsbereichen zu finden sind, macht es Sinn diesen Design-Aspekt anwendungsbereichsübergreifend zu betrachten, anstatt diese jeweils einzeln zu behandeln.

Eine verbreitete Methode zum Management von Varianten und eine typische Design-Aktivität ist die Konfiguration [BAKF04] [HaBR08c] [LaRo08a] [Riit01] [TrGC95]. Dabei geht es um die Gestaltung von Artefakten basierend auf einer Menge vorab definierte Elemente [MiFr89]. Im Prozessmanagement sind die Artefakte die Prozessmodelle bzw. Varianten und die Elemente die diversen Prozesselemente wie Daten, Prozessschritte oder Werkzeuge. Die wesentliche Grundlage bilden *konfigurierbare (Prozess-) Modelle* [BDDK04] [TiSo97] [SaWe97], die mehrere Varianten in einem Modell integrieren. Die Unterscheidungspunkte der Varianten werden konzeptionell explizit als *variation points* (dt. Variationspunkte) erfasst, die als ein Kernpunkt konfigurierbarer Modelle angesehen werden.

Abbildung 1-3 zeigt zwei Varianten des Aufnahmeprozesses (Ausschnitt) einer Klinik. Auf der linken Seite sind sie getrennt modelliert; auf der rechten Seite ist ein integriertes Prozessmodell zu sehen. Die variablen Stellen sind durch die gestrichelte Umrandung hervorgehoben.

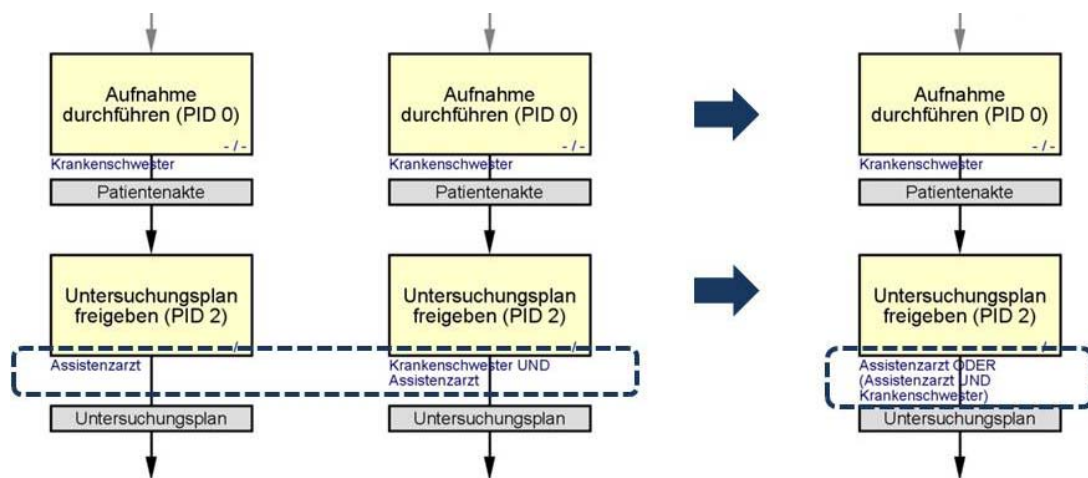


Abbildung 1-3 Varianten und ein integriertes Prozessmodell

Während dies nur ein vereinfachtes Beispiel aus einem Prozessmodell eines Lehrstuhlprojektes mit einem einzigen variation point ist, enthalten Prozesse üblicherweise *mehrere variation points*. Außerdem ist zu beachten, dass diese *unterschiedliche Aspekte* der Prozesse betreffen können. So wie in dem gerade genannten Beispiel der organisatorische Aspekt variiert, können ebenso der funktionale Aspekt oder die Daten eines Prozesses variieren. Dies ist ein essentieller Punkt bei der Definition von Varianten, da nur unter Berücksichtigung aller relevanten Aspekte eine umfassende Beschreibung der Prozesse

möglich ist [CuKO92] [JaBu96]. Darüber hinaus existieren *Abhängigkeiten* zwischen den Auswahlmöglichkeiten verschiedener variation points. So kann die Rollenwahl in dem obigen Beispiel im weiteren Prozessverlauf die Ausführung eines zusätzlichen Prüfschrittes beeinflussen (hier nicht abgebildet). Die Anzahl der Kombinationsmöglichkeiten wird durch die Definition von Abhängigkeiten eingeschränkt und auf eine sinnvolle bzw. erlaubte Menge begrenzt.

Nach diesen Anforderungen gestaltete Prozessmodelle ermöglichen es eine große Menge an Informationen in ein und demselben Prozessmodell abzubilden und bieten eine integrierte Sichtweise auf den Sachverhalt. Doch sind die genannten Punkte nicht immer bzw. nicht vollständig in den existierenden (konfigurierbaren) Modellierungssprachen gegeben.

Werden die variation points eines konfigurierbaren (Prozess-) Modells entschieden, wird daraus ein Teilprozessmodell, eine Variante, abgeleitet. Dazu werden die nicht mehr relevanten Varianten aus dem Prozessmodell gelöscht. Die resultierenden Prozesse werden zur Ausführung oder auch zu anderen Einsatzzwecken wie der Analyse, verwendet.

Bei umfangreichen Prozessmodellen ist zumeist eine *hohe Anzahl* an variation points unter Berücksichtigung der *Abhängigkeiten* aufzulösen, was die Gefahr birgt, dass der für die Gesamtentscheidung verantwortliche Stakeholder schnell den Überblick verliert kann. Dies gilt umso mehr, wenn alle variation points noch vor Beginn der eigentlichen Ausführung aufgelöst werden sollen. Möglicherweise fehlen an dieser Stelle noch *entscheidungsrelevante Informationen*, die erst mit der Ausführung bekannt werden. Es können *essentielle Eigenschaften* bzgl. der Variante gefordert sein; es muss garantiert werden, dass diese zum Beispiel nicht durch die vorherige Entscheidung anderer variation points nicht mehr Bestandteil des finalen Prozessmodells sind.

Die genannten Punkte zeigen, dass nicht nur die Abbildung der Varianten in einem integrierten Prozessmodell, sondern auch deren Konfiguration eine Herausforderung darstellen. Es ist zu klären, was genau Gegenstand der Konfiguration sein soll, um ein entsprechendes Modellierungskonzept entwickeln zu können. Darüber hinaus gilt es, nicht nur die Korrektheit der entstehenden Modelle zu garantieren, sondern dem Anwender auch eine methodische Unterstützung bei dem Entscheidungs- bzw. Konfigurationsprozess an die Hand zu geben.

Die Problematik der konfigurierbaren Prozessmodelle ist insgesamt als ein anwendungsfallübergreifendes Beispiel für das (Prozess-) Designs zu sehen. Es kann ideal dafür verwendet werden, die Notwendigkeit einer expliziten Anforderungsanalyse und der daraus sich gegebenenfalls ergebenden Anpassung oder Neudefinition einer Modellierungssprache und die dazugehörigen Methoden zu veranschaulichen. Die im Rahmen dieser Arbeit entstandene Modellierungssprache ist bereits prototypisch anwendbar.

1.2 Problemstellungen und Zielsetzungen

Trotz der regen Forschungsarbeit (siehe zum [LDHM11], [HaBR10] oder [BDDK04]) sind *konfigurierbare Prozessmodelle* noch nicht als „Stand der Technik“ anzusehen. Es ist eine Vielzahl nicht zufriedenstellender Lösungsansätze vorhanden, zum Beispiel bzgl. der Modellierung von Abhängigkeiten zwischen variantenbezogenen Entscheidungen oder die Ableitung einer Variante aus einem integrierten Prozessmodell.

Nicht nur die spezielle Problematik der Prozessvarianten kann auf einer höheren Abstraktionsstufe, der *Designphase* des Prozesslebenszyklus im Allgemeinen, betrachtet werden, so dass sich auch für die Phase des Designs konkrete Problemstellungen und Zielsetzungen ableiten lassen. Der Schwerpunkt liegt auf der methodischen Vorgehensweise der Designphase.

1.2.1 Prozessdesign

Im Bezug auf die Designphase wurden folgende Problemstellungen und Zielsetzungen identifiziert, die sich in entsprechenden Beiträgen dieser Arbeit widerspiegeln:

- **Problemstellung (1): Unzureichende Durchführung der Designphase vor der (Geschäfts-) Prozessmodellierung**
Die Designphase beschränkt sich derzeit bei vielen Prozessmanagementprojekten auf die Auswahl gegebener Prozessmodellierungssprachen sowie vorhandener Werkzeuge. Auf Grund ihrer festgelegten Eigenschaften und Funktionalität decken diese jedoch je nach Anwendungsgebiet nicht immer alle Anforderungen ab. Es können nicht alle wesentlichen Aspekte der Prozesse in den damit erstellten Prozessmodellen dargestellt werden bzw. nicht mit ausreichender Eindeutigkeit und Präzision. Die individuelle Entwicklung oder das sog. Customizing von Prozessmodellierungssprachen (und Werkzeugen) wird kaum vorgenommen. Dies wirkt sich negativ auf die nachfolgenden Phasen, insbesondere die der Modellierung, aus.
- **Zielsetzung: Konzeptionelle und methodische Erweiterung der Designphase durch das Einbeziehen der Entwicklung und/ oder Anpassung von Prozessmodellierungssprachen**
Ziel ist es, den Anwender methodisch derart zu unterstützen, dass er sich nicht auf die Auswahl aus gegebenen Prozessmodellierungssprachen und -Werkzeugen beschränkt, sondern gegebenenfalls diese auch anwendungsbezogen gestaltet.
- **Beitrag der Arbeit: Konzeptionelle Erweiterung einer existierenden Methodik für die Modellerstellung um den Aspekt des Designs**
Die vorhandene Modellierungsmethodik sieht die Erstellung von Modellen auf verschiedenen Hierarchieebenen vor und differenziert dabei zwischen Definition und Verwendung einer Prozessmodellierungssprache. Bisher wird fast nur die Ebene der

Modellierung wie sie in Kapitel 1.1.1 erläutert wurde genutzt, ohne den Aspekt des Designs zu berücksichtigen. Letzteres wird nun explizit spezifiziert und bezieht auch die Definition/Anpassung von Prozessmodellierungssprachen mit ein. Durch die Integration in ein und dasselbe methodische Rahmenwerk kann eine enge methodische Kopplung der beiden Phasen garantiert werden.

Der methodische Aspekt der Designphase soll noch weiter konkretisiert werden. Dies gilt insbesondere für die zu Beginn der Designphase durchzuführende Anforderungsanalyse.

- *Problemstellung (2): Unzureichende Durchführung der Anforderungsanalyse im Bezug auf die (Geschäfts-) Prozessmodelle*

Um in einem Prozessmodell alle wesentlichen Aspekte eines Anwendungsfalls umfassend und adäquat realisieren zu können, sind eine individuelle Anforderungsanalyse und nachfolgende Umsetzungsspezifikation essentiell. Sie werden jedoch in den bereits genannten Anwendungsdomänen vor allem im Hinblick auf die Definition/Anpassung einer Prozessmodellierungssprache selten durchgeführt.

- *Zielsetzung: Ziel- bzw. umsetzungsorientierte Aufbereitung der Anforderungen*

Im Hinblick auf die adäquate und vollständige Erfassung der Anforderungen sollen dem Anwender klar definierten Fragen gestellt werden, mit denen die einzelnen Anforderungen an eine Prozessmodellierungssprache, an die Prozessmodelle und an die Prozessausführungen identifiziert und klar voneinander getrennt werden können.

- *Beitrag der Arbeit: Strukturierung der Anforderungen gemäß der in (1) verwendeten hierarchischen Modellierungsmethodik*

Die Erweiterung der Modellierungsmethode bzgl. der Designphase mit ihrer Unterscheidung nach Definition und Verwendung einer Prozessmodellierungssprache wird auf die Teilphase der Anforderungsanalyse herunter gebrochen.

1.2.2 Konfigurierbare Prozessmodelle

Folgende Problemstellungen, Zielsetzungen und Beiträge lassen sich bzgl. des Themas Variantenmanagement herausarbeiten:

- *Problemstellung (3): Fehlende Ausdruckstärke der Prozessmodellierungssprachen*

Die variablen Stellen (variation points) integrierter Prozessmodelle sind bei der Verwendung konventioneller Prozessmodellierungssprachen oft nicht eindeutig bzw. explizit zu erkennen; bei den konfigurierbaren Prozessmodellierungssprachen liegt der Schwerpunkt auf der Darstellung variabler Kontrollflüsse, während die Variabilität im Bezug auf andere Aspekte, wie Daten, Organisationen oder Werkzeuge vernachlässigt wird [LDHM11]. Zudem ist die Möglichkeit zur Begrenzung des möglichen Variantenraums nur unzureichend bis gar nicht möglich. Dies reduziert die Ausdruckstärke der Sprachen erheblich.

- **Zielsetzung: Holistische Darstellung der tatsächlich realisierbaren Prozessvarianten**

Es soll eine strukturierte und kompakte Darstellungsform des Variantenraums definiert werden. Dabei muss die Variabilität bzgl. beliebiger Aspekte eines Prozessmodells explizit erfassbar und eine Einschränkung des Variantenraums auf eine konsistente und sinnvolle Menge möglich sein.

- **Beitrag der Arbeit: Konzeption eines generischen, konfigurierbaren Datenmodells zur aspektorientierten Erfassung der Variabilität von Prozessen**

Für die holistische Abbildung der tatsächlich realisierbaren Prozessvarianten wird ein generisches Datenmodell basierend auf einer speziellen Baumstruktur, den sog. mereologischen Graphen [Hümm04], konzipiert. Der Ansatz abstrahiert von einer konkreten Prozessmodellierungssprache wie BPMN oder EPK. Das Datenmodell enthält explizit aspektorientierte variation points zur Darstellung der Variabilität der Prozesse. Eine Begrenzung des Lösungsraums wird durch die explizite Modellierung von Abhängigkeiten realisiert.

Die Erstellung ausdrucksstarker, konfigurierbarer Prozessmodelle reicht jedoch nicht aus. Schließlich muss aus diesem Prozessmodell noch die gewünschte Variante abgeleitet werden. Es hat sich gezeigt, dass dies nicht immer ganz leicht ist. Folgendes kann diesbezüglich festgehalten werden.

- **Problemstellung (4): Schwer überschaubare und kontrollierbare Entscheidungssituation bzgl. einer Variante**

Aufgrund der hohen Komplexität der Entscheidungssituation (mehrere variation points, verschiedene Perspektiven, Abhängigkeiten, besonderen Wünschen, fehlenden Informationen) kann der Stakeholder bei der Auswahl seiner gewünschten Variante schnell überfordert sein und den Überblick verlieren. Wird die Konfigurationsentscheidung auf einen fixen Zeitpunkt beschränkt, ist es sehr wahrscheinlich, dass die final konfigurierte Variante nicht dem gewünschten Ergebnis entspricht.

- **Zielsetzung: Ausarbeitung einer Methode um die Stakeholder durch einen flexiblen und variablen Konfigurationsprozess leiten zu können**

Die Konfiguration soll für den Stakeholder überschaubar und kontrollierbar sein. Er soll die Entscheidungen über die einzelnen variation points dann treffen können, wenn er die dazu notwendigen Informationen hat. Es muss garantiert werden, dass die finale Variante eine korrekte bzw. erlaubte Lösung aus dem Variantenraum darstellt und alle für den Stakeholder essentiellen Eigenschaften aufweist.

- **Beitrag der Arbeit: Konzeption der Methode eines gestuften Konfigurationsprozesses**

Die Gesamtentscheidung für eine Variante soll in überschaubare Teilentscheidungen, die ein oder mehrere variation points abdecken, zerlegt werden. Deren

Auflösungsreihenfolge kann individuell vom Stakeholder festgelegt werden, wobei es sogar möglich sein soll, Entscheidungen bis zur Ausführungszeit hinauszuzögern. Der Stakeholder kann sich somit schrittweise seiner gewünschten Variante nähern, indem mit jeder Teilentscheidung die zu Beginn große Menge an Varianten bis hin zur gewünschten Lösung reduziert wird. Bei jeder einzelnen Konfigurationsstufe können sowohl die Korrektheit des aktuellen Modells überprüft als auch die Folgen der jeweiligen Entscheidung aufgezeigt werden, so dass gegebenenfalls noch frühzeitig Änderungen an der bis dahin durchgeführten Konfiguration möglich sind. Eine Benutzeroberfläche soll nicht entwickelt werden, sondern allein das Konzept des internen Konfigurationsmechanismus.

1.3 Publikationen

Die durchgeführten Forschungsarbeiten führten zu folgenden Publikationen:

Prozessmanagement

- Meerkamm, S.: *The Concept of Process Management in Theory and Practice - A Qualitative Analysis*. 1st. Int'l. Workshop on Empirical Research in Process Management (ER-BPM '09) Ulm, Germany (2009).
- Faerber, M., Meerkamm, S., Schneider, T., Jablonski, S.: *Qualitative Prozessanalyse klinischer Prozesse - Vorgehen und Erfahrung*. arthritis + rheuma 29 (2009) 353-359.
- Faerber, M., Meerkamm, S., Schneider, T., Jablonski, S.: *The ProcessNavigator - Flexible Process Execution for Product Development Projects*. Int'l. Conference on Engineering Design (ICED '09), Stanford (CA), USA (2009).

Design

- Jablonski, S., Meerkamm, S.: *An Extended Interpretation of Process Design and Modelling*. Int'l. Conference on Design Science Research in Information Systems and Technologies (DESRIST '09), Philadelphia (PA), USA (2009).
- Jablonski, S., Meerkamm, S.: *Process Design as Basis for Comprehensive Process Modeling*. Int'l. Conference on Design Science Research in Information Systems and Technologies (DESRIST '10), St.Gallen, Switzerland (2010).
- Meerkamm, S., Jablonski, S.: *A Meta-Model-Framework for Structuring the Requirement Analysis in Process Design*. Int'l. Conference on Design Science Research in Information Systems and Technologies (DESRIST '11), Milwaukee, WI, USA (2011).

Varianten – Prozesskonfigurator

- Meerkamm, S.: *Configuration of Multi-Perspectives Variants*. 1st. Int'l. Workshop on Reuse in Business Process Management (rBPM 2010), Hoboken, NJ, USA (2010).

- Meerkamm, S., Jablonski, S.: *Configurable Process Models: Experiences From A Medical And An Administrative Case Study*. 19th European Conference on Information Systems - ICT and Sustainable Service Development (ECIS '11), Helsinki, Finland (2011).
- Meerkamm, S.: *Staged Configuration of Multi-Perspectives Variants based on a Generic Data Model*. 2nd Int'l. Workshop on Reuse in Business Process Management (rBPM 2011), Clermont-Ferrand, Frankreich (2011).
- Meerkamm, S.: *The Process Configurator*. Int.'l. Journal for Business Process Integration and Management (IJBPI), accepted for publication (2012).

1.4 Aufbau der Arbeit

Die vorliegende Arbeit umfasst neun Kapitel. Sie lässt sich schematisch in insgesamt drei Teile gliedern (Abbildung 1-4): einer Einführung in das Prozessmanagement und methodische Grundlagen in Kapitel 1 bis Kapitel 4, einem theoretischen-konzeptionellen Teil in Kapitel 5 sowie einem anwendungsbezogenen Teil in Kapitel 6.

In Kapitel 2 wird eine Einführung in das Thema *Prozessmanagement* gegeben, die auf einer eingehenden Literaturrecherche basiert. Darüber hinaus werden die Ergebnisse einer eigens durchgeführten qualitativen Studie präsentiert.

In Kapitel 3 wird speziell auf die beiden Phasen *Design* und *Modellierung* im Rahmen des Prozessmanagements eingegangen. Es erfolgt eine Abgrenzung der Begriffe und eine Einführung in die methodischen Grundlagen der beiden Phasen. Kapitel 4 stellt aus Projekten des Lehrstuhls für Angewandte Informatik IV an der Universität Bayreuth ermittelte Anforderungen bzgl. Design und Modellierung vor. Dies leitet letztendlich über zu den Ausarbeitungen von Kapitel 5. In diesem wird konzeptionell ein methodisches Rahmenwerk für die Designphase vorgestellt.

Kapitel 6 fokussiert die *Designphase im Hinblick auf die Modellierung sowie Konfiguration* von Prozessvarianten. Das Konzept eines Prozesskonfigurators wird vorgestellt wobei der Schwerpunkt auf dem dazu entwickelten Datenmodell und dem Konfigurationsprozess liegt. Einblick in die ersten Ansätze der Implementierung können ebenfalls gegeben werden.

Kapitel 7 bilden den Abschluss der Arbeit. Es werden eine Zusammenfassung sowie ein Ausblick gegeben.

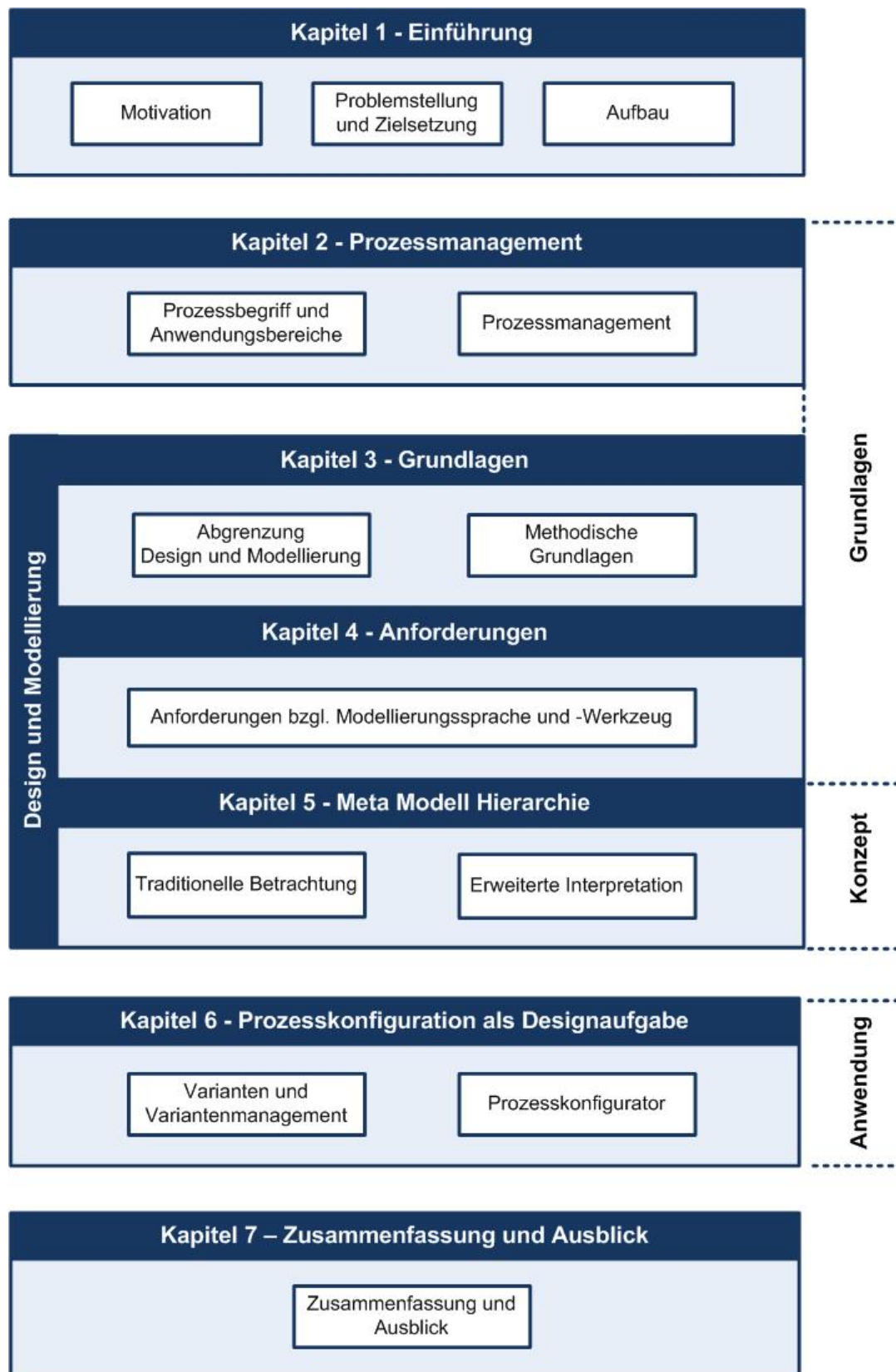


Abbildung 1-4 Schematischer Aufbau der Arbeit

Kapitel 2

Prozessmanagement – Stand in Forschung und Praxis

Wie bereits in Kapitel 1 erläutert, ist die Bedeutung von Prozessen für die Steuerung und Optimierung von Abläufen in der Organisation in den letzten Jahren zunehmend erkannt worden [Dave93] [HaCh93] [Hamm01] [Hamm90] [Hamm97] [SmFi03]. Auch verschiedene Studien konnten zeigen, dass sich die Idee des Prozessmanagements zu einem etablierten Management-Ansatz in Theorie und Praxis entwickelt hat, um ein Unternehmen zu führen und zu organisieren (siehe zum Beispiel [ArMa97], [EHLB95], [GaGr07], [Hung06] oder [WoHa10]). Der Fokus liegt auf dem zielorientierten Management von Zeit, Qualität und Kosten. Dabei gilt es strategische wie operative (Unternehmens-) Ziele zu erreichen und die Effizienz der Abläufe zu erhöhen. Der Ansatz ist somit als ein Fundament der wirtschaftlichen Entwicklung der Organisation und der Wirtschaft als Ganzes anzusehen [Meer07].

Ein Blick in die Literatur zeigt: Es existieren viele verschiedene, zum Teil voneinander abweichende, wenn nicht sogar widersprüchliche Ansätze des Prozessmanagements [vAHW03] [ZaSi95]. Dem gegenüber stehen die Ansätze der Praxis bzw. die in der Praxis umgesetzten theoretischen Ansätze. In diesem Kapitel werden Einblicke in die Theorie und Praxis gegeben sowie Gemeinsamkeiten und Unterschiede aufgezeigt. Ziel ist es, das Problemfeld der Designphase zu konkretisieren, welches im weiteren Verlauf der Arbeit behandelt wird. Für ein besseres Verständnis wird zunächst eine Einführung in den Prozessbegriff gegeben und exemplarisch in ausgewählte Anwendungsbereiche eingeführt. Für einen umfassenden Überblick werden Aspekte des Prozessmanagementansatzes erläutert.

2.1 Der Prozessbegriff

Der Begriff „Prozess“ leitet sich ursprünglich von dem lateinischen „procedere“ ab, übersetzt „vorgehen“ oder „vorrücken“ [Mitt95]. Die heutige Definition des Begriffs *Prozess* ist durch die Verwendung in verschiedensten Anwendungsgebieten jedoch sehr viel spezifischer. Auch im Bereich der (Wirtschafts-) Informatik hat sich in den letzten Jahrzehnten eine Reihe von Definitionen entwickelt, die je nach Verwendung variieren. Im Folgenden werden Definitionen kurz vorgestellt und diskutiert.

Gaitanides [Gait83] liefert folgende, fundamentale Definition:

Prozesse sind inhaltlich abgeschlossene Erfüllungsvorgänge, die in einem logischen inneren Zusammenhang stehen.

Gaitanides geht auf zwei Hauptcharakteristika von Prozessen sein: Mit der Durchführung eines Prozesse ist das Erreichen eines Ziels verbunden, was mit „abgeschlossene Erfüllungsvorgänge“ allgemein beschrieben wird. Die Vorgänge stehen dabei in einer sinnhaften Beziehung, das heißt „logischem ... Zusammenhang“, zueinander.

Konkretere Angaben sind in der Definition von Davenport zu finden [Dave93]:

A process is simply a structured, measured set of activities designed to produce a specified output for a particular customer or market. [...]. A process is thus a specific ordering of work activities across time and place, with a beginning, an end, and clearly identified inputs and outputs: a structure for action.

Die Beziehung bzw. der Zusammenhang der Vorgänge (hier „activities“) wird hier näher beschrieben; nach dieser Definition erfolgt eine *Strukturierung* der einzelnen Aktivitäten (lateinisch „structura“ = ordentliches Zusammenfügen“) und zwar *im Bezug auf Raum und Zeit*. Die Eckpfeiler der zu definierenden Struktur, des nach Gaitanides „abgeschlossenen Erfüllungsvorgangs“, sind die klar zu definierenden Inputs und Outputs. Mit „Kunde“ und „Markt“ wird auch explizit der Empfänger des *Outputs* genannt. Dies zeigt, dass Prozesse nicht isoliert von ihrer Umwelt zu betrachten sind.

Eine weitere Definition, die die Aktivitäten noch näher charakterisiert, liefern Schwarzer und Krcmar [ScKr95]:

Prozesse können definiert werden als die Transformation von Objekten durch vor- und/oder nebengelagerte Aktivitäten eines oder mehrerer Menschen oder Maschinen in Raum und Zeit. Das Ziel ist dabei immer die Erreichung einer vorgegebenen Leistung.

Hier wird zum einen der Begriff der Transformation eingeführt. Dies bedeutet, dass mit der Durchführung der Aktivitäten gemäß einer festgelegten Struktur eine *Veränderung des Objektes* in Raum und Zeit erfolgt. Mensch und Maschine werden explizit als konstituierende Elemente eines Prozesses genannt. Wie bei der Definition von Davenport werden die Prozesse und ihre Aktivitäten nicht isoliert von ihrem Umfeld betrachtet werden können, sondern es werden Werkzeuge eingesetzt und Verantwortliche zugewiesen. Übereinstimmend mit den vorgehenden Definitionen gilt es einen Output, die „vorgegebene Leistung“, zu erzeugen. Schwarzer und Krcmar erwähnen in ihrer Definition außerdem explizit vor- und/oder nebengelagerte Aktivitäten. Sie geben damit einen Hinweis auf die gängige Unterteilung von Prozessen in verschiedene Prozessklassen. Grundlegend wird dabei zwischen primären, wertschöpfenden Prozessen und sekundären, unterstützenden Prozessen unterschieden [ScSe08].

Die Definition von Schwarzer und Krcmar ist bereits sehr umfassend, doch kann sie noch weiter präzisiert werden. Die Prozessklassen sowie der Begriff der Transformation führen zu

dem Begriff der *Geschäftsprozesse* [Gada05] [HaCh93] [JMPW93] [RuBr95] [ScSe08] [ScZi96]. Diese stellen aus der betriebswirtschaftlichen Perspektive eine Untermenge der Prozesse im Allgemeinen dar. Als Beispiele sind Produktentwicklungs- oder Entstehungsprozesse [Ehrl07] sowie medizinische Prozesse [Hell02], wie sie in Krankenhäusern bei der Behandlung von Patienten zu finden sind, zu nennen. Überschreitet bei einem Transformationsprozess der Output den Input wertmäßig, wird dies als *Wertschöpfung* bezeichnet. Im Produktionsprozess stellt das gefertigte Produkt die Wertschöpfung dar; im Kontext der medizinischen Prozesse ist unter Wertschöpfung die Verbesserung des Gesundheitszustandes des Patienten zu verstehen.

Eine entsprechende Prozessdefinition ist bei Hammer und Champy [HaCh93] zu finden:

[Geschäftsprozesse sind eine] Menge von Aktivitäten, für die eine oder mehrere unterschiedliche Inputs benötigt werden und die für den Kunden ein Ergebnis mit Wert erzeugen.

Als zusätzlicher Fokus im Vergleich zu den vorherigen Definitionen ist hier die Kundenorientierung zu nennen. Damit werden jedoch unterstützende Prozesse, die keinen direkt sichtbaren Nutzen für den Kunden haben, wie es zum Beispiel bei verwaltungstechnischen Prozessen der Fall ist, vernachlässigt. In Anlehnung an [Ste99] sollen die Geschäftsprozesse und unterstützenden Prozesse daher unter dem Begriff *Anwendungsprozesse* zusammengefasst werden.

Für die vorliegende Arbeit soll der Begriff des *Anwendungsprozesses* folgendermaßen definiert werden:

Ein Anwendungsprozess umfasst beliebige Aktivitäten, die in einem logischen und zeitlichen Zusammenhang stehen. Dabei ist es das Ziel aus ein oder mehreren Inputs, über verschiedene Zwischenergebnisse, einen Output zu erzeugen bzw. eine Aufgabe zu erfüllen. Die Aktivitäten werden von ein oder mehr Rollen unter Verwendung von entsprechenden Werkzeugen bzw. Hilfsmitteln ausgeführt.

2.2 Anwendungsbereiche des Prozessbegriffs

Wie bereits in Kapitel 2.1 bei der Definition des Prozessbegriffs erwähnt, kommen *Anwendungsprozesse* in den unterschiedlichsten Bereichen vor (Produktentwicklung, Medizin, Verwaltung, ...). Trotz der Unterschiede sind die gleichen Methoden und Konzepte anwendbar. In diesem Abschnitt sollen exemplarisch Anwendungsbereiche vorgestellt werden, um einen tieferen Einblick in dieses Themengebiet zu bekommen. Die Auswahl der Anwendungsgebiete leitet sich ab aus Projekten des Lehrstuhls für Angewandte Informatik IV.

Folgende Bereiche werden behandelt:

- ***Produktentwicklungsprozesse***

Im Rahmen des von der Bayerischen Forschungstiftung geförderten Forschungsverbunds „FORFLOW“ ([MePa06], [MePa08]) galt es einen sogenannten Prozessnavigator zur Prozess- und Workflowunterstützung von Produktentwicklungsprozessen zu erforschen, zu entwickeln und prototypisch umzusetzen. Als Grundlage galt es ein Prozessmodell zu erstellen, das alle für die Produktentwickler notwendigen Informationen abbildet.

- ***Medizinischer Prozesse***

Ziel dieses Projektes war die Einführung sog. Klinischer Behandlungspfade für die medizinische Behandlung von Patienten am Klinikum in Fürth. Dabei erfolgte eine Ist-Analyse der im Klinikum vorhandenen medizinischen Prozesse und deren Dokumentation (siehe auch [FMSJ09]).

- ***Verwaltungstechnische Prozesse an der Universität***

Im Hinblick auf die von der Universitätsleitung der Universität Bayreuth angestrebten System-Akkreditierung galt es die Prozesse im Bereich Studium und Lehre zu Dokumentationszwecken zu erfassen (siehe [UnBa11]).

Es folgt eine kurze Definition und Beschreibung der Anwendungsbereiche. Dabei wird auch auf die Eigenschaften der Prozesse im Hinblick auf Modellierung (und Ausführung) eingegangen.

2.2.1 Produktentwicklungsprozess

Der Produktentwicklungsprozess beginnt mit der ersten Idee zu einem Produkt und den davon abgeleiteten, in einem sogenannten Pflichtenheft festgehaltenen Anforderungen. Der Prozess endet mit der Einführung des Produktes auf dem Markt [Ehrl07] [PaBe97] [ScSe08]. Üblicherweise wird der Prozess in die Phasen des

- *Planens*
- *Konzipierens*
- *Entwerfens und*
- *Ausarbeitens*

unterteilt [PaBe97] [VDI82] [VDI97]. Die Durchführung dieser Phasen erfolgt gemäß verbindlicher *Vorgehensweisen und Vorgehensplänen* [PaBe97]. Sie beschreiben die einzelnen Teilaktivitäten sowie deren In- und Output, wie es zum Beispiel in den VDI-Richtlinien 2221 [VDI93] sowie 2222 [VDI82] [VDI97] zu lesen ist. Sie sind als Richtschnur oder Vorgehenshilfe, gemäß Müller [Müll90] als Prozessmodelle zu verstehen, nicht jedoch als Vorschrift (Beispiel siehe Abbildung 2-1). Sie dienen vor allem dazu das Vorgehen rational zu beschreiben und die Vorgänge transparent zu machen.

Dokumente sind vor allem im Hinblick auf die Zwischenziele (Meilensteine) relevant sowie bzgl. der Endergebnisse zur Dokumentation der Vorgänge. Zu Beginn muss das sogenannte Pflichtenheft spezifiziert werden, das die Anforderungen festhält. Deren Umsetzung erfolgt unter Berücksichtigung diverser Vorschriften und Richtlinien, über die informiert werden muss. Darüber hinaus gilt es Werkzeuge zu definieren (zum Beispiel 3D-CAD-Werkzeuge zur Erstellung von Konstruktionszeichnungen bzw. Datensätzen), die den Produktentwickler bei seiner Arbeit unterstützen und gleichzeitig Informationsbasis für die spätere Entwicklung sind.

Der Produktentwicklungsprozess ist durch viele iterative (Teil-) Prozesse charakterisiert, das heißt es handelt sich nicht um eine starr sequentielle Vorgehensweise. Angestoßen werden sie oftmals durch viele individuelle Denkprozesse. Durch methodisches Vorgehen sollen die *Iterationsschleifen* in Zahl und Umfang möglichst klein bleiben. Die Annäherung an die Lösung erfolgt schrittweise [Ehrl07] [PaBe97]. Die Aktivitäten können nicht immer in der Reihenfolge ausgeführt werden, wie es im Prozessmodell beschrieben ist. Hierbei spielen die *kreativen Prozesse* eine entscheidende Rolle, da der Zeitpunkt einer Ideenfindung nicht festgelegt werden kann. Auch neue Lösungswege sind noch nicht im Prozessmodell abgebildet, sollten aber während der Ausführung in den Gesamtprozess integriert werden können. Der Produktentwicklungsprozess ist außerdem geprägt von einer Vielzahl parallel ablaufender Abschnitte, die in gegenseitiger Wechselwirkung zueinander stehen und somit koordiniert werden müssen. Das Ende einer größeren Etappe wird üblicherweise durch einen Meilenstein gekennzeichnet, an dem es definierte Zwischenziele des jeweiligen Entwicklungsprojektes zu erreichen gilt.

Bei der Umsetzung der Vorgehenspläne in realen Abläufen erfolgt eine Vermischung der Vorgaben aus den Prozessmodellen mit dem *individuellen Denkprozess* der ausführenden Personen [PaBe97]. Die Denkprozesse besitzen einen sehr hohen Innovationscharakter. Der Anteil sogenannter kreativer Prozesse in den Produktentwicklungsprozessen ist sehr hoch [Remu02]. Sie können im Voraus jedoch nicht geplant bzw. in Prozessmodellen festgehalten werden.

Die Bereitstellung der Informationen kann gut durch IT-Systeme unterstützt werden, was sich auf die Datenverfügbarkeit und den Informationsstand der Prozessbeteiligten positiv auswirkt. Eine automatische Dokumentation der Ergebnisse ist damit eng verbunden.

Fazit

In der Produktentwicklung wird die Idee der Prozesse als grundlegendes Konzept zur Strukturierung von Entwicklungsvorhaben verwendet. Die dokumentierten Vorgehensweisen dienen zur Unterstützung der konkreten Ausführung, um zielorientiert zu einem qualitativ guten Entwicklungsergebnis zu kommen. Meilensteine entlang des Prozessverlaufs definieren (Zwischen-) Ergebnisse bzw. Ziele und dienen bei der Durchführung des Produktentwicklungsprozesses die Qualität und den Erfolg sicher zu stellen. Bei der

Ausführung müssen zusätzliche Vorkommnisse und die aktuell vorliegenden Bedingungen, die nicht im Modell festgehalten werden können, berücksichtigt werden, so dass eine flexible Anpassung der Abläufe möglich sein muss.

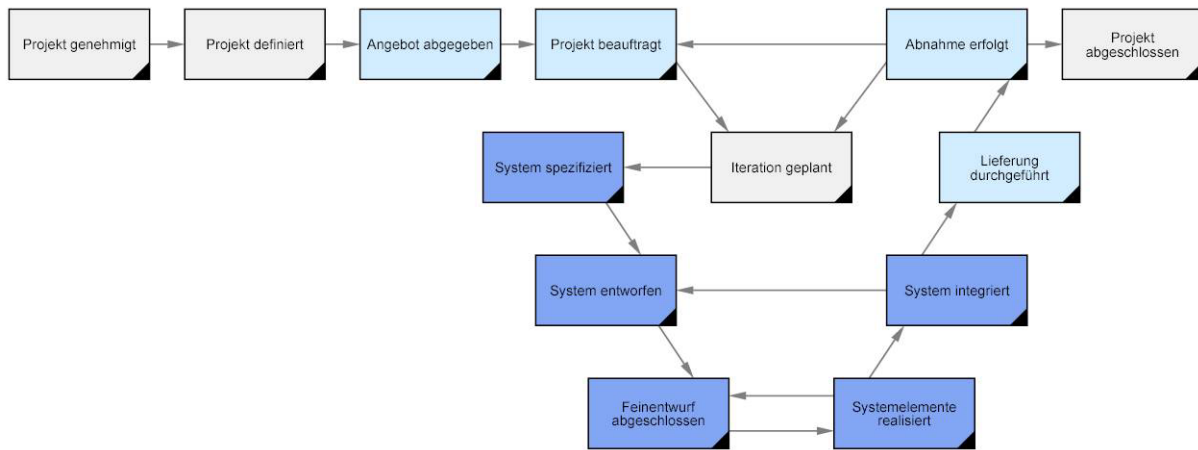


Abbildung 2-1 Beispiel – Produktentwicklungsprozess aus [Meer07b]¹

2.2.2 Medizinischer Prozess

Die Adaption des Prozessgedankens im Bereich der Medizin wird in der Literatur als *Klinischer Pfad* bezeichnet [Hell02]. Die Begriffe *Clinical Pathway*, *Geplanter Behandlungsablauf (GBA)* und *Patientenpfad* können dabei synonym verwendet werden. Nach [Hell02] und [Opit04] wird der *Klinische Pfad* wie folgt definiert:

Ein klinischer Pfad ist ein einzigartiger, berufsgruppenübergreifender Behandlungsablauf auf Evidenz-basierter Grundlage (Leitlinie), der Patientenerwartung, Qualität und Wirtschaftlichkeit gleichermaßen berücksichtigt“

Klinische Pfade stellen eine standardisierte Beschreibung dessen dar, was in einer Klinik üblich ist. Sie werden in der Regel an die lokalen Verhältnisse eines Klinikums angepasst [Hell02], so dass sie die vorhandenen Ressourcen optimal nutzen können. Es herrscht nicht unbegründet die Angst vor einer sogenannten „Koch-Buch-Medizin“, die dem medizinischen Personal die Handlungsfreiheit nimmt [Hell02] [Opit04]. Im eigentlichen Wortsinn sowie unter betriebswirtschaftlichem Gesichtspunkt wird Standardisierung unter anderem als eine Vereinheitlichung von Verfahrensweisen und Abläufen [Broc93] [Gabl10] verstanden. Prinzipiell sollte jedoch der Patient als Individuum mit seiner Krankheit im Vordergrund stehen. Davon unabhängig ist die Etablierung Klinischer Pfade nicht für alle klinischen

¹ Die Erstellung der Beispiele in den folgenden drei Unterkapiteln erfolgte mit der Modellierungssprache AOPM und dem Modellierungswerkzeug i>pm (siehe Kapitel 3.3.1.4).

Abläufe sinnvoll, da der Aufwand den zu erwartenden Nutzen deutlich übersteigen würde [HSMR02].

Die Klinischen Pfade stellen eine prinzipielle Vorgehensweise bzw. Leitlinien auf Basis Evidenz-basierter Grundlagen dar. Die Abläufe sind in der Realität jedoch sehr viel variabler und von vielen unvorhergesehene Ereignissen oder nicht vorher bedachten Konstellationen bzgl. des Gesundheitszustandes eines Patienten oder der Verfügbarkeit von Ressourcen geprägt. Dies stellt eine entscheidende Herausforderung für die Handhabung dieser Prozesse dar.

Teils enthalten die Abläufe eine Vielzahl von Einzelschritten bei denen die Reihenfolge nicht entscheidend ist, die aber auch nicht parallel ausgeführt werden müssen bzw. können. Ein Beispiel dafür wäre der Bereich der Pflege. Oft muss sie innerhalb eines bestimmten Zeitfensters durchgeführt werden (zum Beispiel nach der Einweisung, vor bzw. nach einer Operation). Diese Schritte lassen sich somit mit einer Checkliste vergleichen, deren Anwendung das Ziel einer Vollständigkeitskontrolle (hier die Durchführung der Schritte) verfolgt (siehe dazu zum Beispiel [Meil05]).

Zumeist sind an medizinischen Prozessen sowohl verschiedene Organisationseinheiten (medizinisches Personal, Patienten, Verwaltung, etc.) als auch unterschiedliche Geräte und Systeme (Röntgengerät, CT, KIS, etc.) beteiligt [DaRK00] [LBHH05], was die Komplexität der Prozesse wesentlich erhöht.

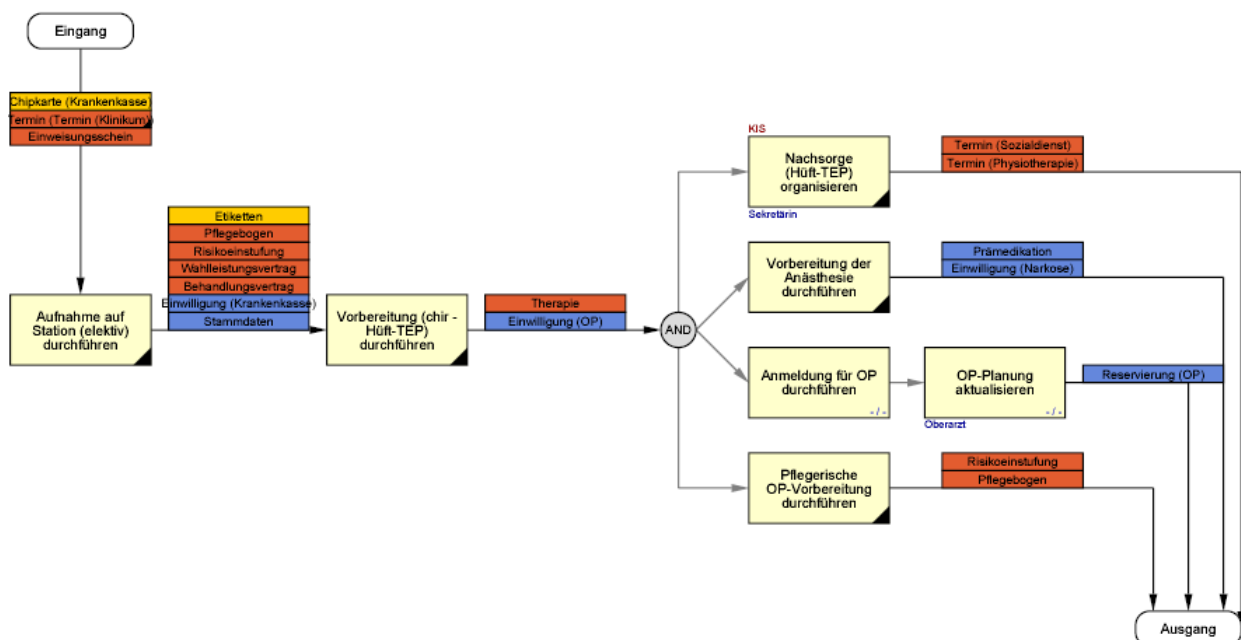


Abbildung 2-2 Beispiel – Medizinischer Prozess [eigene Quellen]

Die Bereitstellung der Informationen kann gut automatisiert und durch IT-Systeme unterstützt werden, was sich auf die Datenverfügbarkeit und den Informationsstand der Prozessbeteiligten positiv auswirkt. Eine automatische Dokumentation der Ergebnisse ist damit eng verbunden. Zum Teil lässt sich dies direkt mit der Durchführung von Teilschritten koppeln, wenn dort technische Systeme, beispielsweise ein EKG, verwendet werden. Viele funktionale Schritte könne aber nur manuell ausgeführt werden, wie zum Beispiel „Anamnese durchführen“, was mit nicht zu vermeidenden Medienbrüchen verbunden ist.

Fazit

Auch in der Medizin wird die Idee der Prozesse als strukturierendes bzw. steuerndes Konzept verwendet. Die Steuerung erfolgt vor allem durch Verwendung von Klinischen Pfaden, welche medizinische Leitlinien abbilden. Die Klinischen Pfade dienen als Grundlage für die Durchführung der medizinischen Prozesse. Da deren Definitionen ein hohes Maß an Allgemeingültigkeit besitzen, muss bei deren Verwendung eine Anpassung gemäß dem individuellen Anwendungsfall erfolgen, wie es auch bei der Produktentwicklung der Fall ist.

2.2.3 Verwaltungstechnischer Prozess an der Universität

Innerhalb des deutschen Hochschulsystems wird immer häufiger von dem "Unternehmen Universität" [Sieb08] [GrvH00] oder Universität als "Wissenschaftsunternehmung" [GrvH00] gesprochen. Vorbilder dieses Paradigmenwechsels sind vor allem die amerikanischen Universitäten [GrvH00]. Auch diese müssen ihr Ziel der „Wissenschaftlichkeit“ wirtschaftlich und effizient erreichen, um im Wettbewerb zwischen den verschiedenen Universitäten bestehen zu können. Als Produkte der Universität sind Erkenntnisse und Wissen aus der Forschung sowie die Absolventen verschiedener Fachrichtungen aus der Lehre zu nennen.

Prozesse, die in dieser Art von Unternehmen ablaufen, werden üblicherweise (vgl. dazu zum Beispiel [Sinz97], [TeUI09] oder [Fähn09]) unterteilt in:

- *Kernprozesse Studium und Lehre*
- *Kernprozesse Forschung sowie*
- *Verwaltungs- bzw. Dienstleistungsprozesse*

Die Aktivitäten des *Kernprozesses Studium und Lehre* sind auf die universitäre Ausbildung der Studenten gerichtet. Mit den Schwerpunkten Bewerbung und Zulassung, Prüfungsverwaltung, Studierendenverwaltung und Veranstaltungsmanagement wird dies zusammengefasst unter dem Begriff „Student-Life-Cycle“. Dabei ist eine starke Interaktion zwischen den Studenten und den universitären Organisationseinheiten zu beobachten. Während sich dies auf die Durchführung eines Studiengangs bezieht, sind außerdem die Teil-Kernprozesse für die Einrichtung sowie Aufhebung eines Studiengangs zu nennen. Diese sind von gesetzlichen Vorschriften, zum Beispiel dem Bayerischen Hochschulgesetz (BayHSchG) oder den Vorgaben der Kultusministerkonferenz, geprägt und stark dokumentenlastig.

Im *Kernprozess Forschung* sollen Forschungsergebnisse erarbeitet werden, die dann der Öffentlichkeit und Wirtschaft, aber auch den Studenten zugänglich gemacht werden sollen.

Die *Verwaltungs- und Dienstleistungsprozesse* sind auf die Kernprozesse ausgerichtet und komplett darin integriert. Beispielhaft zu nennen wären die Personalverwaltung, die Gebäude- und Anlagenverwaltung oder das Bibliothekswesen. Sie sollen so gestaltet werden, dass die Durchführung der Hauptprozesse möglichst effizient unterstützt wird.

Zur Selbstdokumentation und/oder im Hinblick auf die seit dem Beginn des Bologna-Prozesse [BuBF11] durchzuführenden Akkreditierungsmaßnahmen zur Überprüfung des internen Qualitätssicherungssystems einer Hochschule im Bereich Studium und Lehre, werden diese Prozesse in Form von Prozesshandbüchern beschrieben.

Obwohl viele verwaltungstechnische Prozesse zu finden sind, die nach vorherrschender Meinung meist als sehr starr und nach festen Ablaufschema durchgeführt werden, spielt aus der Erfahrung des Projektes heraus auch hier die Flexibilität der Ausführung eine große Rolle. Dennoch gibt es viele Abläufe, die dem klassischen Workflow-Konzept [Jab195] [JaBu96] entsprechen, das heißt automatisiert ausgeführt werden können. Eine Herausforderung ist es Schnittstellen zwischen diesen beiden Arten von Prozessen (flexibel und starr) zu realisieren, wenn sie gemäß der Definition im Prozessmodell direkt aufeinanderfolgen.

Die Prozesse erfordern ansonsten generell eine geringe Werkzeugunterstützung, wichtig ist es vor allem die Rollen richtig zu definieren, damit die Verantwortlichkeiten geklärt sind. Eine hohe Dokumentenlastigkeit ist auffällig; es werden für die einzelnen Schritte meist viele Unterlagen benötigt und dann auch viele erstellt.

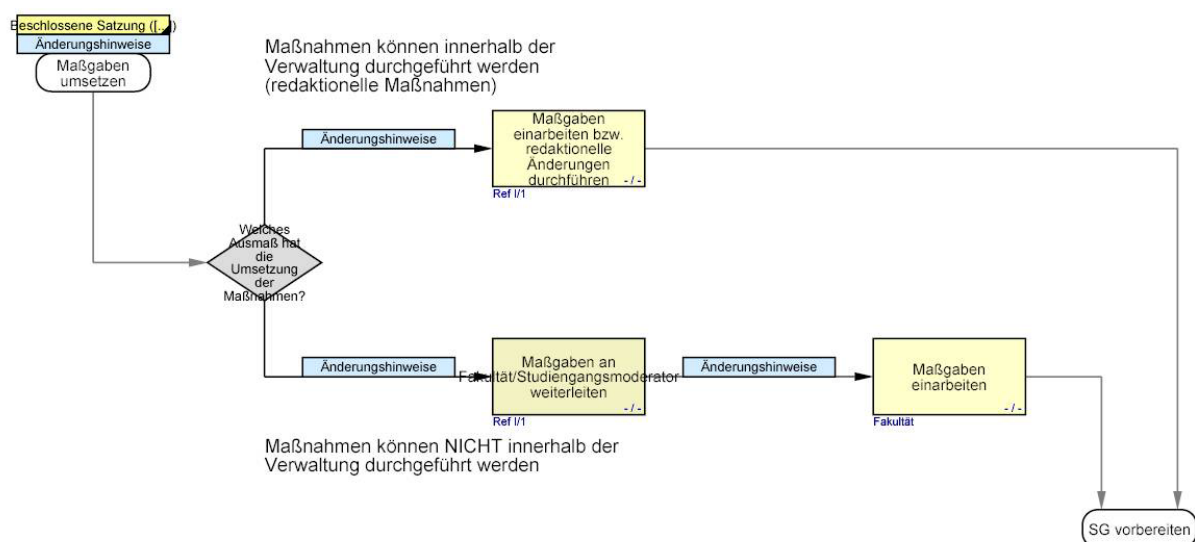


Abbildung 2-3 Beispiel – Universitätsbezogener Prozess [UnBa10]

Fazit

Die Umsetzung des Prozessgedanken kann auch bei den Abläufen innerhalb einer Universität vorgenommen werden. Dies erfolgt vor allem zur Schaffung von Transparenz mittels der Dokumentation der universitären Vorgänge. Bei diesen Prozessen ist es vor allem entscheidend In- und Output der einzelnen Schritte zu definieren. Die damit entstehenden Output- und Input-Beziehung bestimmen den Prozessablauf. Im Vergleich zu den in den vorherigen Kapitel 2.2.1 und 2.2.2 vorgestellte Anwendungsbereichen sind bei der Durchführung der Prozesse weniger Anpassungen bzgl. der Abläufe nötig, da die Abläufe besser planbar sind.

2.2.4 Fazit

Obwohl die drei so eben erläuterten Bereiche auf den ersten Blick wenig mit einander gemeinsam haben, kann die Idee der Prozesse, sei es zur Analyse der Vorgänge oder zur Steuerung der Prozesse und den dabei verwendeten Ressourcen, eingesetzt werden.

Ein erster Schritt zur konkreten Anwendung der Prozessidee ist die Dokumentation, d.h. zur Modellierung, der jeweiligen Vorgänge. Diese können mit Hilfe des Konzeptes der Prozesse entsprechend strukturiert werden. Die Dokumentation dient zudem der Schaffung von Transparenz bzgl. der Abläufe und zur Vermittlung von wichtigen dazugehörigen Informationen, wie zum Beispiel die medizinischen Leitlinien (Medizin) oder der Zuordnung von Verantwortlichkeiten.

Bei allen drei Anwendungsbereichen können Prozessmodelle als eine Art Ablaufplan für die tatsächliche Durchführung verwendet werden. Das Ausmaß an Anpassung bzgl. aktuell vorliegenden Kontextbedingungen ist je nach Anwendungsbereich unterschiedlich; bei den medizinischen Prozessen ist es zum Beispiel sehr hoch, bei den verwaltungstechnischen Prozessen eher niedriger.

2.3 Prozessmanagement

Im Folgenden soll in das Thema Prozessmanagement eingeführt werden. Aus der theoretischen wie auch aus der Perspektive der Praxis wird der Status Quo erläutert. In beiden Fällen wird kein Anspruch auf Vollständigkeit erhoben.

- **Stand in der Forschung:** Es erfolgt eine Darstellung der konsolidierten Ergebnisse aus einer Literaturrecherche.
- **Stand in der Praxis:** Mittels eines Überblicks über diverse empirische Studien, unter anderem einer selber ausgeführten, wird ein Einblick in die Praxis geben.

2.3.1 Prozessmanagement – Stand in der Forschung

Die Aufgaben des Prozessmanagements werden in der Literatur üblicherweise in einem so genannten Prozesslebenszyklus zusammengefasst und erläutert. Wie schon beim Prozessbegriff sind viele unterschiedliche Definitionen zu finden. Da es den Rahmen der Arbeit sprengen würde jeden einzelnen Ansatz detailliert zu präsentieren, soll hier eine konsolidierte Version der Ansätze in [Allw05], [BeKR05], [SBMS06] sowie [Wesk07] erläutert werden. Abbildung 2-4 gibt einen Überblick. Es werden die typischen Inhalte und Aufgaben der einzelnen Phasen des Prozesslebenszyklus erläutert.

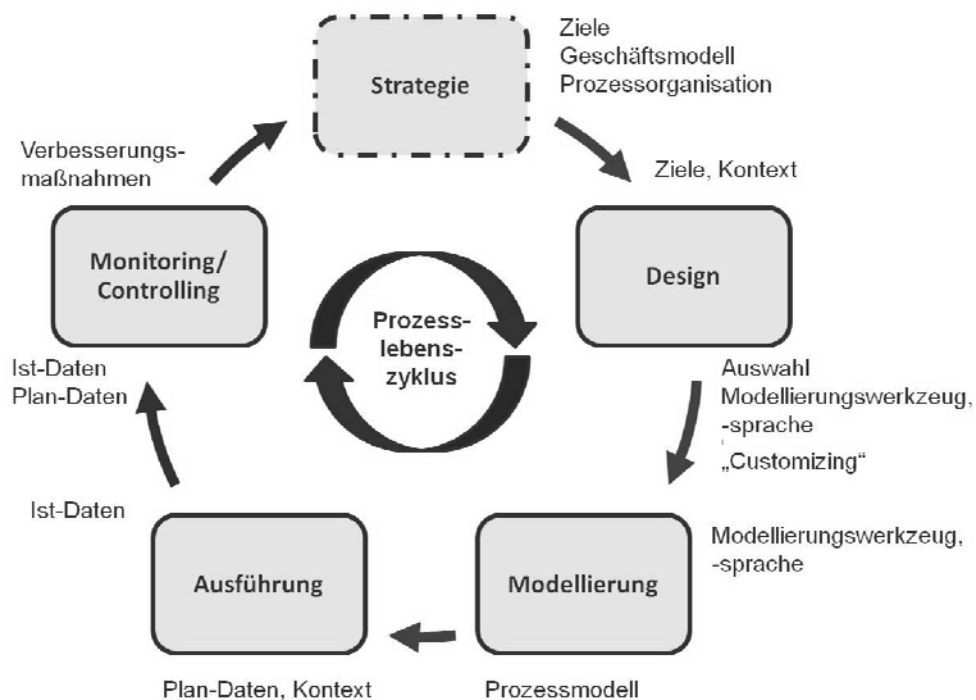


Abbildung 2-4 Prozesslebenszyklus

Design

In der Designphase geht es um die (Neu-) Gestaltung bzw. beim wiederholten Durchlauf des Prozesslebenszyklus auch um die Optimierung der Prozesse [Allw05] [BeKR05] [SBMS06] [Wesk07]. Ausgelöst werden kann dies durch unterschiedliche Faktoren. Nach [SAKR05] wären beispielhaft zu nennen:

- (1) Durch eine Überarbeitung der Abläufe soll in neues, höheres Leistungsniveau erreicht werden [SAKR05], da die aktuellen Prozessparameter und deren Werte nicht mehr zu zufriedenstellenden Ergebnissen führen. So sollen zum Beispiel die Entwicklungskosten gesenkt oder die Produkteinführungszeit (time-to-market) verkürzt werden.

- (2) Beim Einsatz neuer Technologien, zum Beispiel beim Übergang von einem 2D-CAD zu einem 3D-CAD-Werkzeug in der Produktentwicklung, sind für die Installation des Werkzeuges zusätzliche Schritte notwendig oder Verantwortlichkeiten müssen angepasst werden.
- (3) Die Entwicklung einer jungen Organisation, die zunächst noch viele Vorgaben braucht – zu einer reifen Organisation, die eher selbstorganisiert arbeitet, kann die Überarbeitung der Prozesse hervorrufen.

Die Phase des Designs kann dabei in folgende drei Hauptabschnitte unterteilt werden:

- In einem *Analyseschritt* werden zunächst alle formalen und inhaltlichen Anforderungen an die Prozesse, deren Modellierung und Ausführung gesammelt [Allw05] [BeKR05] [SBMS06] [Wesk07]. Vor allem die Anforderungen bezüglich einer Modellierungssprache und deren Funktionalität sowie das entsprechende Modellierungswerkzeug sollen spezifiziert werden. Dieser Schritt ist mit besonderer Sorgfalt durchzuführen, da im Folgenden nur das umgesetzt werden kann, was vorher definiert, das heißt bekannt war.
- Unter Berücksichtigung der strategischen Vorgaben werden die *Umsetzungsmaßnahmen bzgl. der Anforderungen* definiert [Allw05] [BeKR05]. Dies erfolgt im Hinblick auf den sich anschließenden Lebenszyklus. Es gilt die Entscheidung bezüglich einer Modellierungssprache und deren Funktionalität sowie des entsprechenden Modellierungswerkzeugs zu treffen. Kann eine gegebene Modellierungssprache gewählt werden, schließt sich das sogenannte *Customizing* [BeKR05] [Wesk07], die Konfiguration der Modellierungselemente gemäß den gewünschten Anwendungsfeldern sowie die Spezifikation von Modellierungskonventionen an. Andernfalls wird eine komplett neue Modellierungssprache oder –Werkzeug entwickelt.

Es ist bekannt, dass gerade die Anforderungsanalyse und die sich anschließende Spezifikation möglicher Lösungen zur Erfüllung der gewünschten Funktionalität eine große Herausforderung darstellen [DaSh96] [VeCo94].

Die Vorgehensweise deckt sich prinzipiell mit Abbildung 1-2. Wie bereits in Kapitel 1.1.2 erläutert, beschränkt sich jedoch die Gestaltung auch hier auf die Auswahl und/oder Anpassung existierender Modellierungssprachen, während die Neudefinition außer Acht gelassen wird. Somit ist davon auszugehen, dass bei der hier erläuterten Vorgehensweise nicht immer alle relevanten Aspekte eines Prozesses erfasst werden können.

Modellierung

Die Modellierung beinhaltet die Erstellung eines Prozessmodells, das heißt die Abbildungen eines Prozesses aus der realen Welt [Allw05] [BeKR05] [Wesk07]. Es stellt eine visuelle Beschreibung dar, wie Unternehmen ihre Arbeit ausführen [Dave05].

Ein Prozessmodell enthält typischerweise folgende Aspekte:

- *Aktivitäten* und die dazugehörige *Datenfluss-* bzw. *Kontrollflusslogik*² [CuKO92] [Dave05] [GeHS95] [JaBu96]
- Informationen zu *Dokumenten und sonstigen Informationen, organisatorischen, operationalen bzw. IT-Ressourcen* [GeHS95] [JaBu96] [Sche00] [TsJu95]
- *weitere Artefakte* wie zum Beispiel externe Stakeholder oder Metriken zur Erfassung von Leistungsparametern [Sche00] [Sche94] [TsJu95]

Zur Erstellung des Modells wird die im Design ausgewählte und/oder angepasste Modellierungssprache verwendet. Darüber hinaus gilt es die jeweiligen Modellierungskonventionen zu berücksichtigen (siehe zum Beispiel [MeRA10] oder [BeRU00] als allgemeingültige Beispiele).

Abschließend sollte eine Validierung der Modelle (zum Beispiel durch Workshops, Simulation) erfolgen, um sicherzustellen, dass das Modell auch wirklich die gewünschten Informationen enthält und keine Missverständnisse bzgl. des Prozessablaufs vorliegen [Allw05] [BeKR05] [Wesk07]. Die Prozessdesigner und -modellierer müssen sich bewusst sein, dass die Qualität der Modellierungsergebnisse entscheidend von Eigenschaften und Funktionalität der verwendeten Modellierungssprache und den dazugehörigen Werkzeugen abhängig ist. Es können die gewünschten Informationen in einem Modell nur dann eindeutig abgebildet werden, wenn Modellierungssprache und Werkzeug dazu auch die entsprechende Eigenschaften und Funktionalität besitzen [ReMe07].

Ausführung

Nun gilt es die Modelle mit oder ohne Systemlösungen in die Praxis umzusetzen, das heißt zu verwenden³ [Allw05] [BeKR05] [Jabl10] [SBMS06] [Wesk07]. Dies kann auf unterschiedliche Art und Weise erfolgen:

- *mit IT-Unterstützung*: Es erfolgt die Verwendung eines Informationssystems zur (teil-) automatisierten Ausführung der Prozessmodelle bzw. der darin definierten Prozessschritte. Dem Anwender werden die nächsten auszuführenden Schritte angezeigt inklusive der notwendigen Dokumente. Dabei kann es sich um sogenannte Workflow-Management-Systeme [JaBu96] handeln, bei denen die Prozessausführung exakt nach dem zugrundegelegten Prozessmodell erfolgt; der Einsatz erfolgt zum Beispiel bei Verwaltungsprozessen. Darüber hinaus gibt es Systeme, die eine flexible Ausführung ermöglichen (siehe zum Beispiel [FMSJ09] für Prozesse aus der

² In der Literatur wird meist von einem *Kontrollfluss* gesprochen. Die Ablaufreihenfolge der Aktivitäten ergibt sich jedoch in fast allen Fällen aus der Input-Output-Beziehung der Aktivitäten bzw. dem dazwischen liegenden *Datenfluss*.

³ Der Begriff *Verwendung* kann weiter und flexibler gefasst werden als der Begriff *Ausführung*. Mit letzterem wird üblicherweise eine IT-Unterstützung assoziiert, die sich streng an dem vorgegebenen Prozessmodell orientiert (siehe [Jabl10]).

Produktentwicklung). Die Integration weiterer Applikationen ist bei beiden Ansätzen möglich.

- *ohne IT-Unterstützung*: Hier werden die Prozesse bei ihrer Verwendung zum Beispiel manuell durchgeführt, wie es bei pflegerischen Maßnahmen in einer Klinik (Patient waschen) der Fall ist. Die Prozessausführenden wissen entweder wie sie die Prozesse auszuführen haben, oder aber die Prozessmodelle liegen in Form von Handbüchern vor. Darüber hinaus gibt es Prozesse, die im Kopf der Mitarbeiter stattfinden, wie zum Beispiel die Entwicklung kreativer Ideen im Rahmen der Produktentwicklung.

Während der Verwendung müssen oftmals Änderungen am Ablauf oder dazugehörigen Aspekten, zum Beispiel die Zuordnung von Ressourcen zu Schritten, vorgenommen werden [BeKR05]. Diese Änderungen sind notwendig, da sich oft während der Verwendung von Prozessmodellen Bedingungen ergeben, welche zur Modellierungszeit nicht vorhersehbar bzw. nicht bekannt waren. Je anforderungsgerechter die bei der Verwendung zugrunde gelegten Prozessmodelle sind, desto besser kann die Verwendung realisiert werden [Agui04]. Dennoch sind gute Modelle allein keine Garantie für eine erfolgreiche Verwendung; die vorliegenden Kontextbedingungen müssen ebenfalls zu dem jeweiligen Anwendungsfall passen.

Monitoring – Controlling

Im Sinne eines integrierten und umfassenden Qualitätsmanagements werden die Prozesse während der Ausführung überwacht [Allw05] [BeKR05] [SBMS06] [Wesk07]: Es gilt Plan- und Ist-Daten miteinander zu vergleichen, wobei zwei Möglichkeiten zur Verfügung stehen.

- *Monitoring*: Bei gravierenden Abweichungen kann im Rahmen des Monitorings sofort und somit frühzeitig durch entsprechenden Maßnahmen in den Prozessablauf eingriffen werden [BeKR05]. Es handelt sich dabei mehr um ad-hoc-Aktionen.
- *Controlling*: Durch eine grundlegende Analyse der Abweichungen im sich der Ausführung anschließenden Controlling können Konsequenzen bezüglich des nächsten Prozesslebenszyklus abgeleitet werden [Allw05] [SBMS06]. Mit den dort definierten Maßnahmen kann bzw. soll verhindert werden, dass die Verantwortlichen beim nächsten Prozessdurchlauf nicht mit denselben Schwachstellen oder Problemen konfrontiert werden.

Vor allem die im Controlling definierten Maßnahmen haben einen wesentlichen Einfluss auf die nächste Designphase, da damit die Anforderungen präzisiert werden können. Das Controlling leitet einen neuen Durchlauf des Prozesszyklus ein, so dass sich der Kreis hier schließt [BeKR05] [SBMS06].

2.3.2 Prozessmanagement – Stand in der Praxis

Empirische Studien der letzten Jahre haben gezeigt, dass der in Kapitel 2.3.1 vorgestellte Prozesslebenszyklus nicht immer direkt in die Praxis übertragen wird bzw. werden kann. Gerade die Umsetzung der Konzepte geschieht oftmals willkürlich und zufällig, wie unter anderem [LeDa98], [BICS07], [RoBr05], [SIBC07] in qualitativen Untersuchungen feststellen konnten. Im Folgenden werden die wichtigsten Punkte ausgewählter (viel zitierter) Studien konsolidiert präsentiert.

Der internationale Schwerpunkt der Studien im Hinblick auf die Teilnehmer hat die Durchführung einer ergänzenden nationalen empirischen Studie auf diesem Gebiet motiviert, um den Status bzgl. des Geschäftsprozessmanagements „direkt vor der eigenen Haustür“ zu ermitteln, deren Ergebnisse hier ebenfalls präsentiert werden. Ziel ist es ein Verständnis zu vermitteln, wie es sich in der Praxis verhält bzw. welche Unterschiede es zwischen Theorie und Praxis gibt.

2.3.2.1 Studien zum Thema Prozessmanagement

Es wird ein Überblick über die Ergebnisse folgender qualitativen Studien gegeben, die sich mit dem Thema *Prozessmanagement* als Ganzes auseinander gesetzt haben:

[AnCh07], [BICS07], [Hung06], [Kuen00], [LeDa98], [RevA05], [SIBC07], [WoHa10] und [WICS07]

Die Studien gehen auf aktuell angewendete Methoden und Konzepte des Prozessmanagements ein und wie sie den Erfolg einer Organisation beeinflussen. [Kuen00] und [RevA05] fokussieren dabei explizit den Einsatz von Workflow-Management-Systemen. Hervorzuheben ist zudem die Studie von [SIBC07], die das Prozessmanagement und die dort eingesetzten Technologien aus Sicht der Anbieter von BPM-Technologien untersucht, während in den übrigen Studien die Information bzgl. der Sicht der Anwender des Prozessmanagements betrachtet werden.

Darüber hinaus wurden aufgrund der Gesamthematik dieser Arbeit weitere Studien bzw. Erfahrungsberichte hinzugezogen, die sich vor allem auf die *Modellierung* konzentrieren:

[DaKo05], [DGRI06], [IGRR09], [IRRG09], [Rose06a] und [Rose06b]

[DGRI06] untersucht wie und zu welchem Zweck Modellierung erfolgt. Während [IGRR09], Nutzen und Vorteile der Modellierung identifiziert (in [DaKo05] weniger explizit zu finden), werden in [Rose06a] und [Rose06b], aber auch in [IRRG09] kritische Aspekte sowie Herausforderungen herausgearbeitet.

Die Auswahl der Studien erfolgte gemäß Themengebiet (Prozessmanagement sowie Modellierung) und Nennung der jeweiligen Journals, Konferenzen oder Workshops in der WI-Orientierungslisten [WKWI08]. Darüber hinaus wurden vier der Studien aufgrund der Angaben in den Referenzen der übrigen Studien hinzugenommen.

Die Erläuterungen orientieren sich gemäß der noch in Kapitel 2.3.2.2 vorzustellenden Studie an folgenden Aspekten:

- *strategisches und operatives Prozessmanagement*
- *Modellierung*
- *Ausführung*
- *Informationstechnologien*

Die genannten Studien folgen nicht explizit diesem Aufbau bzw. haben nicht durchgehend alle aufgelisteten Aspekte erfasst bzw. nicht mit demselben Ausmaß. Daher galt es die Ergebnisse der Studien kritisch zu analysieren, um die relevanten Informationen im Hinblick auf die angestrebte Zusammenfassung extrahieren zu können. Außerdem ist zu berücksichtigen, dass die für die verschiedenen Studien Verantwortlichen jeweils ihre spezifische (subjektive) Sichtweise auf das Thema Prozessmanagement haben, was sich auf die in den Studien gestellten Fragen sowie die Ergebnisanalyse auswirkt.

Strategisches und operatives Management

(Geschäfts-) Prozessmanagement wird sowohl auf der strategischen als auch auf der operativen Ebene praktiziert. Es gilt die Prozesse zu analysieren und darauf aufbauend zu managen, zu kontrollieren und kontinuierlich zu verbessern. Ziel ist es letztendlich die Qualität der zu erstellenden Produkte bzw. allgemein Leistungen zu verbessern [AnCh07] [LeDa98]. Soweit sind diese Ergebnisse mit der in Kapitel 2.3.1 erläuterten Theorie konform.

Demgegenüber haben die Studien jedoch gezeigt, dass es oftmals an der Unterstützung des Managements fehlt, wenn es um die Umsetzung des Prozessmanagements geht [AnCh07] [BICS07] [LeDa98]. Als Ursachen wurden Fehleinschätzungen bzgl. der Bedeutung des Themas Prozessmanagements sowie fehlendes Bewusstsein und Verständnis identifiziert. Gerade auf der *strategischen Ebenen* ist häufig kein gemeinsames Verständnis bzgl. des Begriffs (Geschäfts-) Prozessmanagement vorhanden [WICS07], während auf taktischer Ebene als Übergang zu operativen Ebenen dieses Thema meist gar nicht angesprochen wird. Somit ist der Übergang von der strategischen zur operativen Ebenen unterbrochen [SIBC07].

Auf *operativer Ebene* ist ein Mangel bis hin zu Nichtvorhandensein einer entsprechenden Methodik, von Richtlinien oder allgemeingültigen Praktiken zu beobachten [BICS07] [IRRG09] [SIBC07]. Dies geht einher mit fehlender Ausbildung im Hinblick auf den gesamten Managementansatz. Die Umsetzung geschieht vielmehr willkürlich, zufällig [LeDa98].

Aus diesen Erkenntnissen heraus ist es nicht überraschend, dass viele Unternehmen weiterhin funktionsorientiert, statt prozessorientiert, ausgerichtet sind. Es fehlt dabei vor allem an funktionsübergreifender Koordination sowie Kommunikation [LeDa98]. Somit scheint das Prozessmanagement in der Praxis noch einiges an Nachholbedarf zu haben, was auch die folgenden Abschnitte verdeutlichen.

Modellierung

Die derzeit auf dem Markt vorherrschenden Prozessmodellierungssprachen decken ein sehr breites Anwendungsgebiet ab [DGRI06], so zum Beispiel im Rahmen der Gestaltung von Informationssystemen [DuAH05] oder Service-orientierten Architekturen [Erl05] sowie zum Zweck der Dokumentation der Vorgänge in einem Unternehmen [DaSh90]. Im Einzelnen sind die Modellierungssprachen eher allgemeingültig konzipiert, als dass sie auf einen bestimmten Modellierungszweck ausgerichtet sind [BICS07] [DGRI06] [Rose06a] [Rose06b] [SIBC07]. Die am häufigsten verwendeten Techniken sind

- *ER-Diagramme* (= Methode zur Beschreibung von Objekten und ihren Beziehungen zueinander [Chen76]) in kleineren Unternehmen mit weniger als 100 Mitarbeitern sowie in großen mit über 1000 Mitarbeitern [DGRI06]
- *Flowcharts/ Programmablaufpläne* sowie *UML-Diagramme* in Unternehmen von mittlerer Größe [DGRI06]

Im Bezug auf die am häufigsten verwendeten Werkzeuge ist an vorderster Stelle

- *Visio* (= weitverbreitete Visualisierungs-Software von Microsoft für Windows [MiCo11]) vor allem in mittelgroßen bis großen Unternehmen

zu nennen.

Der Nutzen der Modellierung wird vor allem in dem besseren Verständnis der Prozesse durch visuelle Darstellung gesehen [BICS07] [IRRG09]. Die Darstellung der Prozesse in Form der Prozessmodelle wird gerne zur Dokumentation der Abläufe und Organisationsstruktur eines Unternehmens verwendet, aber auch für die Spezifikation von prozessorientierten Systemen im Rahmen der Softwareentwicklung [DaKo05]. Die Vorteile beziehen sich hauptsächlich auf die operative Ebene sowie die organisatorischen und innerbetrieblichen Bereiche des Prozessmanagements und weniger auf den strategischen Bereich [IGRR09].

Trotz des großen Angebots an Modellierungssprachen ist es dennoch oftmals nicht möglich, die Geschäftsprozesse mit den gewünschten Inhalten und/ oder in der entsprechenden Form strukturiert darzustellen [BICS07] [Rose06a] [Rose06b] [SIBC07]. Es herrscht eine Diskrepanz zwischen dem, was dargestellt werden kann, und dem, was dargestellt werden soll. Die Modellierungssprachen scheinen nicht den Anforderungen des Modellierungskontextes zu entsprechen. Dies erklärt, dass oftmals gar keine Dokumentation vorzufinden ist, da keine Abbildung der Sachverhalte möglich ist [LeDa98]. [SIBC07] berichtet, dass die verwendeten Modellierungssprachen bzgl. der Anwendungsfälle zu allgemeingültig definiert sind. (Die in den Studien genannten Modellierungssprachen werden alle von entsprechenden Organisationen definiert.) Dennoch wird vor allem den Standardisierungsvorhaben im Bezug auf Modellierungssprachen eine hohe Bedeutung zugewiesen [IRRG09]. Es wurde nicht explizit danach gefragt, jedoch scheint keine individuelle Entwicklung oder Anpassung von Modellierungssprachen, wie es gemäß dem

theoretischen Konzept des Prozesslebenszyklus in Kapitel 2.3.1 in der Designphase möglich ist, vorgenommen zu werden. Daher kann auch keine Aussage darüber gemacht werden, warum diese Möglichkeit nicht genutzt wird, obwohl sich die Anwender bewusst sind, dass die vorhandenen Modellierungssprachen nicht immer adäquat sind.

Ausführung

Trotz der allgemeinen Anerkennung des (Geschäfts-) Prozessmanagementansatzes lässt sich dieser im Alltag doch manchmal nur schwer „richtig verkaufen“. Dies ist darauf zurückzuführen, dass die Anwender sehr unterschiedliche Vorstellungen von der Realisierung und dem Output des Management-Ansatzes haben [BICS07] [WICS07] bzw. das entsprechende Bewusstsein fehlt [LeDa98] [WICS07]. Zudem scheint nach [BICS07] [WICS07] vor allem eine Lücke zwischen der Design- und Ausführungsphase vorhanden zu sein; die in der Design-Phase definierten Anforderungen werden nicht konsequent umgesetzt, was Probleme in der Ausführung zur Folge hat. Der Grund dafür liegt vor allem darin, dass in jeder Phase des Prozesslebenszyklus verschiedene Werkzeuge zum Einsatz kommen, die im Hinblick auf eine durchgehende Umsetzung der Anforderungen entlang des Prozesslebenszyklus nicht aufeinander abgestimmt sind. Zudem stehen nur allgemein definierte Modellierungssprachen zur Auswahl, die den individuellen Sachverhalt einer Organisation bzgl. der späteren Ausführung nicht abbilden können.

Um die mit den genannten Punkten verbundenen Schwachstellen bzw. Fehler auszumerzen, müssen entsprechende Maßnahmen eingeleitet werden. [LeDa98] kritisiert, dass aus diesem Grund bei der Ausführung der Prozesse die Durchführung von Verbesserungsmaßnahmen oftmals zu sehr im Vordergrund steht. Die Prozessausführung verläuft also schwerpunktmäßig reaktiv, als dass sie aktiv gestaltet werden kann bzw. wird. Die Probleme werden nicht wirklich an der „Wurzel gepackt“, sondern nur oberflächlich behandelt. Die Motive der Unternehmen, seien es aus zum Beispiel Zeit- oder Kostenaspekten, werden in den Studien nicht ermittelt.

Informationstechnologien

Obwohl die IT (= Informationstechnologie) als ein entscheidender Erfolgsfaktor für das Prozessmanagement angesehen wird [Hung06] bzw. dieser Management-Ansatz als Integrator zwischen IT und der eigentlichen Geschäftstätigkeit fungiert [AnCh07], werden die IT-Systeme oftmals nicht ausreichend auf die Prozesse abgestimmt bzw. umgekehrt. Dies führt dazu, dass die Funktionalität der IT-Systeme nicht den Anforderungen der Prozesse entspricht, was bei der Verwendung der Systeme zu Problemen führt und die Qualität der Prozessergebnisse mindert.

Die im Verlauf des Prozesslebenszyklus eingesetzten Tools stellen Insellösungen dar [Wesk07]. Dies führt dazu, dass aufgrund fehlender Standards Ergebnisse, die in einer Phase mit einem bestimmten System erzielt wurden, nicht nahtlos in eine sich anschließende Phase

übernommen werden können, da die Systeme dazu nicht bzw. nicht ausreichend aufeinander abgestimmt sind. Unter Berücksichtigung dessen was in Kapitel 2.3.1 zu den einzelnen Phasen erläutert wurde, zeigt dies, dass wohl zwischen dem Design und der Ausführung eine direkte Verbindung besteht, die im Prozesslebenszyklus in Abbildung 2-4 nicht explizit berücksichtigt ist. Die Entscheidungen im Design dürfen nicht nur die nachfolgende Phase der Modellierung betrachten, sondern müssen den sich danach fortsetzenden Prozesslebenszyklus im Auge haben. So sollte für die Modellierung eine Modellierungssprache und -werkzeug ausgewählt werden, mit denen es möglich ist Prozessmodelle zu erstellen, die in das Ausführungssystem importiert werden können.

Laut [Kuen00] ist als eine Ursache die mangelnde Einbindung des operativen Managements in die IT-Projekte zu sehen, so dass eine phasen- und fachübergreifende Abstimmung und Diskussion im Hinblick auf die Realisierung eines anforderungsgerechten Systems nicht möglich ist. [Kuen00] bezieht sich in seiner Studie zwar speziell auf Workflow-Systeme, doch kann dies auf prozessorientierte IT-Systeme im Allgemeinen übertragen werden.

Vor der Implementierung eines IT-Systems sollten vorhandene Prozessdefinitionen überarbeitet werden, um IT-Systeme und Prozesse aufeinander abstimmen zu können. [Kuen00] und [RevA05] haben jedoch herausgefunden, dass dies, wenn überhaupt, nur zu einem geringen Ausmaß geschieht und die entwickelten IT-Konzepte meist auf die vorhandene Prozessstruktur angewendet wird. Die Phase des Designs wird somit auch hier nicht umfassend ausgeführt und zur Verfügung stehende Handlungsmöglichkeiten für eine möglichst optimale Realisierung des Prozessmanagementkonzeptes bleiben ungenutzt.

Fazit

Zusammenfassend kann als Ergebnis der Studien festgehalten werden, dass es sich beim Prozessmanagement um einen sowohl auf strategischer als auch operativer Ebene etablierten Managementansatz handelt. Es hat sich gezeigt, dass die prozessorientierte Durchführung der Geschäftstätigkeit zu einer Erhöhung der Qualität des Outputs sowie der Produktivität (vor allem im Hinblick auf die Durchlaufzeit) führt.

Dennoch konnten an verschiedenen Stellen des Prozesslebenszyklus Schwachpunkte identifiziert werden. Hervorzuheben ist, dass bei der Modellierung, sei es zum Zweck der Abbildung der Organisationsstruktur oder im Rahmen der Entwicklung von IT-Systemen, gewünschte und notwendige Inhalte nicht dargestellt werden können bzw. die Modellierer sich auf die gegebenen Aspekte beschränken. In dem Zusammenhang wurde auch bemängelt, dass die IT-Systeme und (die auf der Basis von Modellen erstellt werden) und Prozesse nicht optimal aufeinander abgestimmt sind. Dies betrifft konkret die von den IT-Systemen angebotene Funktionalität. Leider wird in den Studien nicht analysiert, was die Unternehmen daran hindert die individuellen Anforderungen eines Anwendungsfalles auch individuell umzusetzen.

Die genannten Schwachstellen lassen sich mit einer inadäquaten Vorbereitung der Modellierung erklären, was sich auf die Designphase des Prozesslebenszyklus bezieht. Die Designphase ist aufgrund ihrer Position im Prozesslebenszyklus der zentrale Ansatzpunkt für „erfolgreiche“, das heißt effiziente und effektive Prozesse anzusehen, da dort nach Festlegung der Strategie die ersten Grundlagen für den gesamten Prozesslebenszyklus gelegt werden. Die identifizierten Probleme motivieren, in der Designphase nach den Ursachen zu forschen bzw. die Durchführung der Designphase und/oder methodische Vorgehensweis zu optimieren. Damit sollten dann auch Mängel bei der Modellierung oder Ausführung verringert werden können.

Es wird sich zeigen, ob die nun vorzustellenden Ergebnisse der eigenen Studie zu demselben Fazit kommen. Diese werden in dem sich nun anschließenden Abschnitt 2.3.2.2 präsentiert.

2.3.2.2 Eigene Studie zum Thema Prozessmanagement

Die in Kapitel 2.3.2.1 vorgestellten Interviews sind fast ausschließlich international, das heißt mit Interviewpartner außerhalb von Deutschland, durchgeführt worden. Daher wurde mittels einer *qualitativen Untersuchungsmethode* [Lamn95] [Maye06] in Form eines sogenannten *Leitfaden-Interviews* [Lamn95] [Maye06] [Witz00] eine empirische Studie in neun deutschen Unternehmen durchführt, um auch einen Überblick bzgl. des Geschäftsprozessmanagements in Unternehmen direkt „vor Ort“ zu haben und eventuell kulturelle Unterschiede zu identifizieren. Im Folgenden wird zunächst die Untersuchungsmethode näher erläutert, daran schließt sich die Präsentation der Ergebnisse an. Diese Ergebnisse wurden bereits in [Meer09] veröffentlicht.

2.3.2.2.1 Research Design

Eine qualitative Untersuchungsmethode verwendet offene Fragen zur Datensammlung. Meist werden Interviews durchgeführt, dies jedoch mit nur wenigen Teilnehmern. Die zu diskutierenden Themen werden vor dem Interview definiert ohne jedoch eine konkrete Hypothese zu formulieren. Die Reihenfolge in der die einzelnen Themengebiete im Laufe des Interviews angesprochen werden und in welchem Ausmaß, ergibt sich aus dem Verlauf des Gesprächs [Lamn95] [Maye06].

Der verwendete Leitfaden gibt sowohl dem Interview als auch den dabei gesammelten Daten eine Ordnung bzw. Struktur. Dazu folgen die Interviewer der vorab definierten Leitlinie, auch um sicherzustellen, dass alle notwendige Punkte angesprochen werden. Dennoch sind spontane Fragen erlaubt. Es erfolgt inhaltlich eine integrierte Betrachtung von Prozesserfassung (Design, Modellierung) und Prozessverwendung (Ausführung), um Zusammenhänge bzw. Wechselwirkungen aufdecken zu können. Zudem wurden Fragestellungen im Bezug auf die Mitarbeiter als separates Themengebiet definiert; schließlich sind sie es, die den Managementansatz realisieren bzw. „leben“ müssen. Auch die

Informationstechnologien als wesentlicher Bestandteil der heutigen Geschäftswelt werden behandelt. Der detaillierte Leitfaden ist dem Anhang A zu entnehmen. Er entspricht folgendem Aufbau:

- **Bedeutung des Prozessmanagement**
Hier wird zunächst ermittelt, inwiefern das Thema Prozessmanagement überhaupt bekannt ist, um eine Orientierung für den weiteren Verlauf des Interviews zu haben.
- **Rahmenbedingungen**
Die strategischen und operativen Rahmenbedingungen, welche die Basis zur Umsetzung des Managementansatzes darstellen, werden erfragt.
- **Prozesserfassung und – Verwendung**
Zum einen wird ermittelt, wie Identifikation und Definition der Prozesse im befragten Unternehmen erfolgen; zum anderen, wie diese dann umgesetzt werden.
- **Informationstechnologien**
Hier wird die Rolle der Informationstechnologien bezüglich der Prozesserfassung und -verwendung analysiert.
- **MitarbeiterIn**
Letztendlich wird auf die Rolle der Mitarbeiter eingegangen und wie diese mit dem Thema Prozessmanagement konfrontiert werden.

Für die Auswahl der Teilnehmer konnte auf Kontakte des Lehrstuhls durch Forschungsprojekte sowie Kontakte der Universität Bayreuth insgesamt zurückgegriffen werden. Dabei wurde der Fokus auf Mitarbeiter gelegt, die im Bereich des Prozess- sowie Qualitätsmanagements bzw. auf Managementebene der Unternehmen tätig waren. Die potentiellen Teilnehmer wurden zunächst per e-Mail angeschrieben. Nach der dann folgenden telefonischen Kontaktaufnahme zu den sich zur Verfügung stellenden Teilnehmern wurden die Interviews jeweils vor Ort in den Unternehmen durchgeführt. Einen Überblick über die wichtigsten Kennzahlen der Unternehmen gibt Tabelle 2-1. Die Übersicht zeigt, dass die Unternehmen sowohl im produzierenden als auch im dienstleistungsbezogenen Bereich tätig sind. Die Anzahl der Mitarbeiter ist sehr unterschiedlich, von der kleinsten Firma, die 13 Mitarbeiter hat, bis zu den Größten mit mehr als 330.000.

Die Gespräche wurden elektronisch aufgezeichnet, wobei per Hand zusätzliche Notizen erstellt wurden. Nachfolgend wurden die Daten transkribiert. Für die Auswertung der qualitativen Daten wurde das „pragmatische Auswertungsverfahren nach Mühlfeld“ gewählt [Maye06]. Ziel dieses Verfahrens ist es nicht, die Interviews so genau wie möglich zu interpretieren, sondern Problembereiche oder Auffälligkeiten zu identifizieren, um daraus allgemeingültige Aussagen abzuleiten. Der Vorteil zu anderen Auswertungsverfahren ist darin zu sehen, dass die Ergebnisse direkt umgesetzt werden, ohne zum Beispiel den Zwischenschritt einer Paraphrasierung durchführen zu müssen.

Nr.	Beschreibung der Organisation	Rolle des Interviewten	Anzahl der Mitarbeiter	Umsatz in € Jahr
1	Medizintechnik	Manager	40.000	k.A.
2	IT-Dienstleistung	Geschäftsführer	13	k.A.
3	Handel	Manager	170	k.A.
4	Werkzeugbau	Geschäftsführer	380	44 Mio €/Jahr
5	Elektrotechnik	Manager	170	k.A.
6	Automobil	Manager	330.000	108.897 Mio €/Jahr
7	Anlagenbau	Geschäftsführer	40	>10 Mio €/Jahr
8	IT	Manager	100	k.A.
9	Spedition	Controller	1.300	150 Mio €/Jahr

Tabelle 2-1 Übersicht über die befragten Unternehmen

2.3.2.2.2 Ergebnisse

Im Folgenden werden nun die Ergebnisse der Interviewreihe vorgestellt und gleichzeitig ein Vergleich zu den theoretischem Konzept des in Kapitel 2.3.1 vorgestellten Prozesslebenszyklus sowie zu den Ergebnissen der internationalen Studien durchgeführt.

Bedeutung des Prozessmanagements

Innerhalb der befragten Unternehmen ist das Prozessmanagement als Managementtool sowohl auf dem strategischen wie auch auf operativem Level grundlegend akzeptiert. Lediglich eine Firma vertritt eine eher kritische Meinung dazu, was aber wohl den schlechten Erfahrungswerten in Form von schlechten Kennzahlen (vor allem bzgl. des Umsatzes) zuzuordnen ist.

Mehr als zwei Drittel der Unternehmen sehen die Einführung des Prozessmanagementansatzes in engen Zusammenhang mit bzw. als Grundlage für die *Zertifizierung*, vor allem im Hinblick auf die ISO-Zertifizierungen nach ISO 9000 und ISO 9001. Da sowohl die ISO-Zertifizierung als auch das Prozessmanagement ein integraler Bestandteil des Qualitätsmanagements sind, werden sie in einem engen Zusammenhang gesehen. Diese Korrelation wird auch in der Literatur viel diskutiert (siehe zum Beispiel [Allw05], [BeKR05], [CuKO92] oder [ZaSi95]). Neben dem ISO-Standard werden darüber hinaus von fünf der neun interviewten Unternehmen auch weitere Standards erwähnt, wie zum Beispiel die VDI-Richtlinien 2221 [VDI93] und 2222 [VDI82] [VDI97] (Regeln des Vereins Deutscher Ingenieure für die Ingenieurwissenschaft) oder ITIL [OoGC11] (Best

Practice bzgl. der IT-Infrastruktur). Ihre Bedeutung scheint im Zusammenhang mit dem Thema Prozessmanagement jedoch geringer zu sein.

Prozessmanagement wird außerdem als Werkzeug für eine komplette *Reorganisation* des Geschäftsbetriebs verwendet. Dieses Konzept kam bereits 1990 auf (siehe [Hamm90], [DaSh90]) und scheint auch heute noch aktuell zu sein. In zwei Fällen war die Einführung dieses Managementansatzes der „letzte Rettungsanker“ (Nr.4) für die Unternehmen; das Ergebnis der kompletten Umgestaltung war eine drastische Verbesserung des Geschäftsbetriebes. In den anderen Fällen war die Situation weniger dramatisch. Der Ansatz des Prozessmanagements sollte vielmehr zum Zweck der *kontinuierlichen Verbesserung* eingesetzt werden, hier insbesondere bzgl. der internen Kommunikation zwischen den Mitarbeitern und den Schnittstellen zwischen den verschiedenen Arbeitsbereichen bzw. Abteilungen. Ein Erfolg konnte von allen, bis auf eine Ausnahme, bestätigt werden.

Rahmenbedingungen

Strategische Rahmenbedingungen

Die befragten Unternehmen, die Prozessmanagement erfolgreich praktizieren, weisen eine *flache Organisationsstruktur*, in Kombination mit einer *Projektorganisation* für den täglichen Geschäftsbetrieb auf. Dies entspricht dem, was auch in der Literatur allgemein vertreten wird (siehe zum Beispiel [Allw05], [BeKR05]). Die „Richtigkeit“ dieses Organisationstyps wird damit unterstrichen, dass Unternehmen, die unzufrieden mit der Adaption dieses Managementansatzes sind, eine hierarchische Organisationsstruktur mit vielen Ebenen aufweisen.

Wie bereits erwähnt waren, mit Ausnahme von zwei Unternehmen, alle interviewten Unternehmen nach *ISO 9000 und 9001* zertifiziert; einer der weltweit bekanntesten Qualitätsmanagementstandards, der beschreibt, welchen Anforderungen das Management eines Unternehmens genügen muss, um einem bestimmten Standard bei der Umsetzung des Qualitätsmanagements zu entsprechen. Damit wird der enge Zusammenhang zwischen Prozess- und Qualitätsmanagement deutlich. Es sollte erwähnt werden, dass es sich bei den beiden Unternehmen ohne Zertifizierung einmal um ein sehr kleines Unternehmen gehandelt hat, für die Kosten und Nutzen einer Zertifizierung in keinem vertretbaren Verhältnis stehen; im zweiten Fall handelt es sich um ein hochspezialisiertes Unternehmen, für das spezielle Regularien und weniger Standardnormen relevant sind.

Die Idee des prozessorientierten Denkens und Arbeitens dient der *Planung und Steuerung der strategischen Aufgaben* der Unternehmen. Dazu werden zur Darstellung der Kernprozesse Prozessmodelle in Form von „Prozesslandschaften“ oder „Prozesshäusern“, wie es auch in der Literatur beschrieben wird (siehe [Allw05], [BeKR05]), verwendet. Als wesentliche Informationselemente dieser Modelle sind Meilensteine und Arbeitspakete zu erwähnen. Meilensteine werden im Rahmen des Projektmanagements zur Darstellung von (Zwischen-)

Ziel verwendet; Arbeitspakete dienen der detaillierten Aufgabenbeschreibung. Ziel ist es letztendlich auch mit Hilfe der Prozessmodelle die verschiedenen Geschäftsbereiche und ihre jeweiligen Wechselwirkungen abzubilden.

Im Normalfall schließt sich der strategischen Definition des Prozessmanagement-Ansatzes die Operationalisierung der Prozesse an. Nur in einem Fall konnte dieses Konzept nicht beobachtet werden; es handelte sich jedoch um ein sehr kleines Unternehmen, in dem strategische und operative Ebene sehr eng beieinander liegen und weniger streng getrennt sind, als dies bei großen Unternehmen der Fall ist. Die Durchführung des Prozessmanagements auf rein operativer Ebene wird nicht praktiziert; die strategische und operative Ebene sind zwei sich notwendigerweise ergänzende Bereiche, wobei bzgl. der Umsetzung von der strategischen Ebene her begonnen werden sollte, um ein entsprechendes Rahmenwerk vorzugeben.

Konkrete Mängel, Schwachstellen oder Kritikpunkte konnten im Hinblick auf die Gestaltung der strategischen und operativen Rahmenbedingungen bis dahin nicht aufgedeckt werden.

Operative Rahmenbedingungen

Während die internationalen Studien unter anderem ein Mangel an methodischen Vorgehensweisen feststellen konnten (siehe zum Beispiel [WICS07]), ist der nationale Geschäftsbetrieb von Projekten geprägt. Vorgehensmodelle, so zum Beispiel das Wasserfall-Modell [Royc70] oder das V-Modell XT [BuIn06] sind wenig bekannt und deshalb nur wenig im Einsatz. Nur ein Drittel der befragten Unternehmen verwendet sie. Es scheint, dass die Vorgehensmodelle nicht den Anforderungen der Unternehmen entsprechen, Idee und Umsetzung nicht stimmig zueinander sind. Auch weitere Methodiken im Hinblick auf die Umsetzung des Prozessmanagement wurden von den Unternehmen nicht erwähnt. Lediglich einmal wurde in dem Zusammenhang von SCM (= Supply-Chain-Management) [ScSe08] und CRM (= Customer-Relationship-Management) [ScSe08] berichtet. Es ist somit eine Übereinstimmung mit den Ergebnissen der internationalen Studien festzustellen und die theoretischen Konzepte scheinen eine nur unzureichende Umsetzung nach sich zu ziehen.

Ohne Ausnahme betonten alle Unternehmen, dass es nicht allein ausreicht, die Prozesse in Modellen zu definieren und dokumentieren. „Die Prozesse müssen auch in den Köpfen der Mitarbeiter sein“ (Nr.1, Tabelle 2-1). Haben sie eine grundlegende Vorstellung von den Vorgängen, können sie die Prozesse mit der entsprechend vorhandenen Handlungs- und Gestaltungsfreiheit in Kombination mit ihrem eigenen Erfahrungsschatz durchführen. Die übergreifende Koordination erfolgt durch den Projektleiter, der dabei neben dem zu erreichenden Projektergebnis das Gesamtziel des Unternehmens zu berücksichtigen hat. Wichtig ist es, die Prozesse den Angestellten und dem gesamten Geschäftsbetrieb nicht einfach aufzuerlegen. Die Prozessmodelle sollen daher nicht immer 1:1 ausgeführt werden, sondern dienen vielmehr als eine Art Rahmenwerk. Nur in zwei Unternehmen wurde dem gewissermaßen widersprochen und eine strikte Ausführung gefordert. In dem einen Fall

handelte sich um die Abläufe in einer Produktionsanlage; in dem Prozessmodell werden die bei der Produktion gemäß den Vorschriften auszuführenden Schritte abgebildet. Die Einhaltung dieser Regeln ist essentiell für qualitativ gute Produkte. Im anderen Fall wurde das Prozessmodell sehr allgemein bzw. grobgranular im Hinblick auf den funktionalen Aspekt definiert, jedoch sehr genaue Angaben zum Endergebnis gemacht. Die „genaue“ Umsetzung des Prozessmodells war somit bzgl. der definierten Abläufe leicht möglich; entscheidend war, am Ende das richtige Ergebnis zu haben.

Prozesserfassung und -Verwendung

Im Unterschied zu den internationalen Studien wurde zusätzlich zur Modellierung auch die vorausgehende (Teil-) Phase der Prozessidentifikation empirisch untersucht, um auch einen Einblick in die Designphase zu erhalten. Beides wird zusammengefasst unter dem Begriff Prozesserfassung. Die Prozessverwendung bezieht sich auf die Ausführung der Prozesse.

Identifikation + Modellierung

Die Identifikation und Modellierung der Prozesse findet nach Aussage der interviewten Partner in allen Funktions-, bzw. Geschäftsbereichen der Unternehmen statt. Dies zeigt, dass aus dieser Perspektive das Prozessmanagement (bzw. zumindest ein Teil davon) ganzheitlich in den Unternehmen integriert ist.

Unterschiede sind bei den Unternehmen im *Detaillierungsgrad* der Analyse zu finden; nicht nur zwischen den Unternehmen, sondern auch innerhalb eines Unternehmens. Je nach Kontext und Prozessart wird auf den Ebenen der Phasen, der Prozesse oder der Aufgaben analysiert und schließlich modelliert. Während Phasen den funktionalen Aspekt sehr grob umreißen, erfolgt mit Prozessen bzw. Aufgaben eine zunehmend detailliertere Spezifikation. Der Detaillierungsgrad unterscheidet sich zudem von Anwendungsbereich zu Anwendungsbereich. Administrative Prozesse werden weitaus genauer abgebildet als Produktentwicklungsprozesse. Für kleinere Unternehmen muss der Inhalt nicht so detailliert angegeben werden, wie für größere (Grenzwert liegt hier bei Anzahl der Mitarbeiter > 1300); Grund ist die mit der Unternehmensgröße zunehmende Umfang der Abläufe und der dazugehörigen Aspekte (wie zum Beispiel Input- und Outputdaten), die es zu beherrschen gilt. [LeDa98] beurteilt dies hingegen weitaus negativer. Dies bezieht er vor allem darauf, dass oftmals in vergleichbaren Bereichen auf unterschiedlichem Detaillierungsgrad gearbeitet wird. [LeDa98] berücksichtigt nicht, dass noch weitaus mehr Faktoren den Detaillierungsgrad eines Modellierungsprojektes beeinflussen.

Hauptverantwortlich für die Identifikation (und Modellierung) der Prozesse sind die Projektleiter oder (erfahrene) Mitarbeiter, die für das Prozess- oder allgemein Qualitätsmanagements zuständig sind. Die Positionen der Verantwortlichen zeigt, welche Bedeutung dem Prozessmanagement beigemessen wird und, dass das Prozessmanagement

von der Managementebene abwärts her umgesetzt wird. Dies konnte bereits im Kontext der in dieser Interviewreihe erläuterten strategischen Rahmenbedingungen konstatiert werden.

Auf die Frage des *Inhalts der Modelle* wurden folgende Antworten gegeben:

- Als wichtigstes Element wurde die Darstellung von *Aufgaben* genannt, die wiederum in *Teilaufgaben* zerlegt sein können. Die Reihenfolge der Aufgaben wird durch den *Kontrollfluss* abgebildet. Die Festlegung der Reihenfolge aufgrund der Input und Output-Beziehungen war für die Unternehmen in keiner Weise relevant.
- Mit Ausnahme von einem Unternehmen liegt der weitere Fokus auf der Darstellung der mit den Prozessen verbundenen Objekte in Form des jeweiligen *In- bzw. Outputs*. Als Beispiele wurden Dokumente, Templates oder physische Produkte genannt.
- Als weiterer essentieller Bestandteil der Prozessmodelle wurde der *organisatorischen Aspekt* in Form von Rollenangaben genannt, um zu definieren, wer einen Prozessschritt ausführen darf bzw. soll. Lediglich bei den kleinen Unternehmen (Anzahl der Mitarbeiter < 100) wurde dies als nicht notwendig angesehen, da dort eine rollenbezogene Aufgabenverteilung durch die geringe Mitarbeiterzahl nicht als sinnvoll erachtet wird.
- Wie bereits im Abschnitt zu den strategischen Rahmenbedingungen erwähnt, sind *Meilensteine* ein wesentlicher Bestandteil von Prozessmodellen, um Deadlines bzgl. Zeit und Inhalt (= Prozessfortschritt) und letztendlich auch Ergebnisse abbilden zu können.

Es ist festzuhalten, dass all diese Modellierungselemente auch in der gängigen Literatur erwähnt und als relevant erachtet werden (siehe zum Beispiel [CuKO92], [Wesk07]). Anders sieht es bei folgenden Aspekten aus:

- Der *Kostenaspekt* wird nur selten abgebildet, obwohl der Ansatz des Prozessmanagements grundsätzlich mit diesem Parameter in Verbindung gebracht wird (siehe zum Beispiel [BeKR05] [ScSe08]). Einen Grund für das Fehlen konnte nicht ermittelt werden.
- Die Spezifikation von *Werkzeugen und Systemen*, die zur Ausführung eines Prozessschrittes notwendig sind, wie zum Beispiel ein CAD-System oder Software wie zum Beispiel Word, scheint ebenfalls von geringer Bedeutung zu sein. Dies ist überraschend, da vor allem die IT-Systeme bei vielen Unternehmen als wesentlicher Bestandteil des Prozessmanagements angesehen werden [AHMM02] [Allw05] [Wesk07].

Wie auch bei den Studien aus Kapitel 2.3.2.1 geht es insgesamt darum einen Überblick über den Geschäftsbetrieb und den dazugehörigen Hauptaktivitäten zu geben, so dass jeder ein gewisses Grundverständnis über das Was und Wie bekommen kann. Darüber hinaus ist für die Unternehmen vor allem die daten- bzw. informationsbezogene Perspektive von Interesse, wie es auch in der Literatur gesehen wird (zum Beispiel bei [AHMM02], [Allw05],

[CuKO92]). Vor allem die großen Unternehmen mit mehr als 40.000 Mitarbeitern favorisieren eine perspektiven-orientierte Sichtweise, während dies für kleinere Unternehmen weniger relevant zu sein scheint. Keines der Unternehmen erwähnt jedoch individuelle, speziell auf die eigene Organisation zugeschnittene Modellierungsaspekte. Im Hinblick auf die Entwicklung von entsprechenden Informationssystemen stand die Erstellung von Modellen weniger im Vordergrund.

Keines der Unternehmen verwendet ein auf dem Markt vorhandenes offizielles *Modellierungswerkzeug*, wie zum Beispiel Aris Express von der Software AG [Soft11] oder Rational Software Architect von IBM [IBM11]. Vielmehr kommen die verschiedenen Office-Produkte zum Einsatz, sei es Word, Excel, Power-Point oder Front-Page. Dies deutet darauf hin, dass der derzeitige Status Quo bzgl. der Funktionalität, Inhalt und Flexibilität der Modellierungswerkzeuge nicht zufriedenstellend ist. Auch [IRRG09] schlussfolgerten in ihrer Studie, dass Modellierungssprachen und –werkzeuge noch besser an die Marktbedingungen angepasst werden können. Die Entwicklung eigener Systeme zur optimalen Anpassung an die jeweils vorliegenden Anforderungen scheint für die in dieser Interviewreihe befragten Unternehmen kein Thema zu sein, auch bei [IRRG09] wird dies nicht erwähnt.

Im Hinblick auf die *Art und Weise der Dokumentation* praktizieren die Unternehmen eine

- graphische (zum Beispiel Flussdiagramm in Visio oder PowerPoint) oder
- textuelle (zum Beispiel mit Word)

Dokumentation. Dabei müssen sich ISO-zertifizierte Unternehmen (nur zwei Unternehmen der Interviewreihe waren es nicht) an die Anforderungen des Standards halten. Was das konkret beinhaltet, wurde nicht spezifiziert. Die Darstellung sollte außerdem möglichst kompakt erfolgen. Das bedeutet, dass sich ein für sich geschlossener Prozess, zum Beispiel der Abrechnungsvorgang beim Kunden, auf ein Blatt beschränken soll.

Es wird großer Wert darauf gelegt, dass die generierten Prozessmodelle in persönlichen Treffen mit den Anwendern nochmals abgestimmt werden. Gerade im Hinblick auf bereits laufende, aktuelle Prozesse dient dies übereinstimmend dem Zweck der kontinuierlichen Verbesserung bzgl. der Prozesse oder auch der gesamten Geschäftstätigkeit. Die Unternehmen sind sich bewusst, dass ein optimales Ergebnis nicht mit einer einmaligen Aktion erreicht werden kann, sondern kontinuierlich und wiederholt stattfinden muss. Zudem ist dies ein Hinweis darauf, dass der in Kapitel 2.3.1 vorgestellte Prozesslebenszyklus wirklich praktiziert wird, was den Ergebnissen der internationalen Studien nicht derart deutlich zu entnehmen war.

Die meisten Firmen, bis auf eine Ausnahme, veröffentlichen die Prozessdokumentationen schließlich im Intranet oder in Form von Prozesshandbüchern.

Prozessausführung

Es reicht nicht aus, die Prozesse zu identifizieren und zu modellieren. Um den Prozessmanagement-Ansatz umfassend umzusetzen, müssen die Prozesse auch ausgeführt, bzw. verwendet werden. In Kapitel 2.3.1 wurde bereits darauf hingewiesen, dass es vorgezogen wird von *Verwendung* zu sprechen, da sonst schnell die rein technische Ausführung impliziert wird und alternative Herangehensweisen in den Hintergrund treten.

Die Ergebnisse der Interviewreihe unterstützen diese Argumentation. So wurden in keinem der befragten Unternehmen Workflow-Management-Systeme, die die automatisierte und informationstechnisch unterstützte Ausführung der Prozesse beinhalten [Jab195] [JaBu96], als zentraler Ansatz für das Prozessmanagement eingesetzt.

Wie bereits im Zusammenhang mit den operativen Rahmenbedingungen für das Prozessmanagement erwähnt (siehe S. 39), ist es für die Unternehmen vielmehr wichtig, dass die Prozesse für eine erfolgreiche Ausführung auch „in den Köpfen der Mitarbeiter“ sein müssen (Nr.1, Tabelle 2-1). Genauer gesagt müssen sie zum einen die prozessorientierte Denkweise verinnerlicht haben und mit den Abläufen des Unternehmens, dem sie zugehören, vertraut sein. Haben sie ausreichend Erfahrung und wird ihnen die notwendige Handlungs- sowie Gestaltungsfreiheit gewährt, können die Mitarbeiter selbständig arbeiten ohne eine exakte Prozessdefinition zu benötigen. Es sollte hier erwähnt werden, dass dieser Aspekt der Prozessverwendung in der Literatur nicht sonderlich explizit thematisiert wird. Prozessmodelle haben dennoch vielfältige Einsatzzwecke:

- Auch wenn Handlungs- und Gestaltungsfreiheit gewährt wird, liefern die Prozessmodelle während der Ausführung die *Grundlage zur Organisation der Geschäftsaktivitäten*. Sie werden quasi als Richtlinien für „eine wert- und zielorientierte Arbeitsweise“ (Nr.7, Tabelle 2-1) verwendet. Sie müssen jedoch stets dem aktuellen Kontext angepasst werden, was eine wesentliche Herausforderung darstellt und auf keinen Fall unterschätzt werden sollte.
- Um die Mitarbeiter optimal auf die Prozessausführung vorzubereiten, werden die Prozessmodelle in Form von Handbüchern auch zu Schulungen/ Trainings verwendet, um die Mitarbeiter mit den Abläufen des Unternehmen vertraut zu machen. Mit den dort erhaltenen Informationen sollte es ihnen möglich sein, ihre Tätigkeiten selbständig, aber dennoch konform mit den Unternehmensrichtlinien auszuführen. Dieses Ergebnis widerspricht den Ergebnissen der internationalen Studien (zum Beispiel [BICS07], [IRRG09], [SIBC07]). Dort wurde herausgefunden, dass es oftmals an einem systematischen Training bzgl. der Prozesse bzw. dem Prozessmanagement-Ansatz als Ganzes fehlt. Dies könnte jedoch schlicht an den kulturellen Unterschieden der Länder liegen, aus denen die jeweiligen Studienteilnehmer kommen.

- In sieben von neun Unternehmen waren die Prozessmodelle auch die Voraussetzungen für Audits oder Zertifizierungen, vor allem ISO-Zertifizierungen. In dem Kontext geben die Prozessmodell externen, für das Audit oder die Zertifizierung zuständigen Organisationen, einen Einblick in die Geschäftstätigkeit, woraufhin diese dann entsprechend beurteilt werden kann. Gerade bzgl. Zertifizierungen wird der Ansatz des Prozessmanagements gerne als Mittel zur Implementierung eines Qualitätsmanagementsystems verwendet. Im Bereich der internationalen Studien ist dazu leider kaum etwas zu finden.
- Werden die Prozessmodelle zu Beginn dazu verwendet, den Mitarbeitern die betriebseigenen Abläufe näher zu bringen, werden sie danach als Kontrollinstrument zum *Abgleich von Soll und Ist* verwendet. Ein Unternehmen (Nr.9, Tabelle 2-1) erwähnte dabei explizit, dass dies gerade im Hinblick auf mehrere Standorte eines Unternehmens von Vorteil sei, da damit eine Vergleichbarkeit zwischen den Standorten und den jeweiligen Abläufen garantiert werden könnte. Die Unternehmen betonten in dem Zusammenhang, dass die Analyse der Prozesse als eine „never ending story“ anzusehen ist. Nach der Identifikation und der Modellierung wird versucht, die Prozesse kontinuierlich zu verbessern. Werden die Unternehmensziele nicht erreicht, erfolgt eine Analyse der Prozesse, um die Schwachstellen, Missverständnisse aufdecken und Maßnahmen identifizieren zu können. Dies entspricht genau dem Konzept wie es in der Literatur vertreten wird (siehe Kapitel 2.3.1) und resultiert in dem dort beschriebenen Prozesslebenszyklus. Auch wenn es viel Disziplin und Selbstkritik erfordert, schätzen die befragten Unternehmen gerade diese Facette des Prozessmanagementansatzes als sehr wertvoll ein.

Insgesamt äußerten sich die Unternehmen positiv im Hinblick auf die Verwendung der Prozessmodelle im Rahmen der Ausführung, da sich die *Qualität der Prozesse sowie ihrer Ergebnisse* erhöhen. Die Prozessdefinitionen und ihre explizite Darstellung bzw. Verfügbarkeit für den Mitarbeiter führen dazu, dass „die Prozesse sauber ausgeführt“ werden (Nr.5, Tabelle 2-1). Transparenz und das Bewusstsein bzgl. der Schnittstellen zwischen den einzelnen Bereichen steigen ebenfalls an. Dies führt auch zu einer Zusammenführung der Mitarbeiter und ihrem Wissen zu Teams. Die Mitarbeiter entwickeln ein breiteres Verständnis bzgl. der Vorgänge, die sie selber bzw. ihre Kollegen ausführen. Prinzipiell wenden die Unternehmen durchgehend einen ganzheitlichen (im Bezug auf den Prozesslebenszyklus) sowie sehr umfassenden Ansatz des Prozessmanagements an.

Informationstechnologien

Struktur/ Aufbau

Im Bezug auf den Aufbau der IT-Systeme kann eine klare Abhängigkeit von der Größe des jeweiligen Unternehmens festgestellt werden. Kleine Unternehmen (Anzahl der Mitarbeiter < 100) operieren mit *einem übergreifenden System*. Größe des Unternehmens und der Umfang

der Geschäftstätigkeit ermöglichen dies. Von den großen Unternehmen hat nur eines ein derartiges System installiert, während die restlichen *mit verschiedenen Anwendungen agieren, die nicht notwendigerweise zu einer übergreifenden Einheit zusammengefügt sind*. Die Gründe für Letzteres sind nachvollziehbar:

- Die IT-Landschaft wächst im Laufe der Zeit mit dem Unternehmen organisch mit. Über die Jahre hinweg kommen aufgrund neuer, geänderter Anforderungen immer wieder neue Anwendungen hinzu, die (auch aufgrund mangelnder Zeit und Ressourcen) als Add-on verwendet werden, ohne sie vollständig in die existierende IT-Landschaft zu integrieren.
- Die Unternehmen arbeiten mit diversen Lieferanten und Kunden zusammen, wobei vor allem Letztere ihrerseits fordern, dass der Lieferant mit dem gleichen System arbeitet wie sie. Sie produzieren die verschiedensten Produkte und Service-Leistungen, wofür die unterschiedlichsten Anwendungen notwendig sind.

Dennoch sprechen sich auch die großen Unternehmen prinzipiell für eine Integration der verschiedenen Anwendungen aus, zum Beispiel auf Basis eines ERP-Systems (= Enterprise Resource Planning). Diese Aussage zeigt, dass die IT-Strukturen aktuell als verbesserungswürdig angesehen werden und die in der Theorie stark fokussierten Konzepte (siehe zum Beispiel [Allw05]) noch nicht in der Realität umgesetzt sind.

Funktionalität

Obwohl die Unternehmen unterschiedlichen Geschäftsfähigkeiten nachgehen, sind die Anwendungen, die sie verwenden, zum Teil ähnlich. Sie können somit als domänenunabhängige Anwendungen angesehen werden. Ein Zusammenhang mit der Größe der Unternehmen konnte dabei nicht festgestellt werden.

- Generell wird das *Microsoft Office Paket* verwendet, wobei ein Unternehmen Excel sogar als seine Hauptanwendung definiert.
- *Administrative Anwendungen* sind bei allen Unternehmen zu finden.
- *Produktdaten- und/ oder Projektmanagementsysteme* sind in fünf von sieben Unternehmen, die in der Fertigungswirtschaft tätig sind, im Einsatz.

Darüber hinaus hat jedes Unternehmen entsprechend seinem Geschäftsfeld spezielle Anwendungen, im Automobilbereich zum Beispiel ein spezielles Hallenplanungssystem.

Obwohl vorher mit einer Ausnahme alle Unternehmen bestätigten, Prozessmanagement zu praktizieren, kommt in keinem Fall ein explizites Prozessmanagementsystem oder eine entsprechende Integrationsplattform wie zum Beispiel die ARIS Plattform [IDSS10b] zum Einsatz. Untermauert durch die Ergebnisse der internationalen Studien (vgl. zum Beispiel [IRRG09]) kann wohl behauptet werden, dass die aktuell auf dem Markt vorhandenen Prozessmanagementsysteme nicht den Anforderungen entsprechen, auch wenn sie konzeptuell bereits gut ausgearbeitet und durchdacht sind (siehe dazu [AHMM02] und

[Allw05] aber zum Beispiel auch [ReRD05]). Dieser Status Quo ist vergleichbar mit dem der Vorgehensmodelle, die kaum bekannt sind, wie auf S. 39 ausgeführt. Selbst wenn also die Mitarbeiter IT-Systeme zur Ausführung ihrer Prozesse verwenden, sind diese beiden Aspekte nicht optimal aufeinander abgestimmt. Die IT stellt somit einen eindeutigen Ansatzpunkt dar, um die Prozessorientierung in den Unternehmen zu verbessern. Dies wird auch in der Literatur bestätigt (siehe [AHMM02], [Allw05]).

Aber die IT sollte nicht als alleiniger, ausschlaggebender Erfolgsfaktor im Hinblick auf die effiziente Realisierung der Prozesse bzw. des gesamten Managementansatzes betrachtet werden. Alle Unternehmen betonten, dass die Interaktion zwischen den Mitarbeitern ein entscheidender Faktor ist. So ist es Aufgabe der Projektleiter, die Teammitglieder mit notwendigen sowie ausreichenden Informationen zu versorgen; ebenso wird die Kommunikation zwischen den Mitarbeitern als wichtig angesehen. Die Tatsache, dass das Thema Kommunikation über das Interview verteilt immer wieder thematisiert wurde und vor allem bei den Fragen, bei denen es eigentlich um technische Aspekte des Prozessmanagement-Ansatzes ging, zeigt die Bedeutung der Kommunikation. Nach [IRRG09] ist die Kommunikation gleichzeitig auch als ein Gewinn des (Geschäftsprozess-) Management-Ansatzes gesehen. Damit hat sich in den letzten Jahren wohl einiges geändert, da [LeDa98] noch über ein Mangel an funktionsübergreifender Koordination und Kommunikation klagt. Die IT ist inzwischen mehr als ein Datenhaltungssystem.

Implementierung

Das bloße Vorhandensein bzw. die rein technische Implementierung der IT-Systeme alleine erhöhen noch nicht die Wirtschaftlichkeit eines Unternehmens. Die Anwendungen müssen für die Mitarbeiter auch zugänglich sein, wie es auch schon [Hung06] bemerkt hat. Damit ist weniger der technische Zugang an sich gemeint, sondern vielmehr die Benutzungsfreundlichkeit (vgl. dazu auch [IRRG09], die dies speziell im Hinblick auf Modellierungssysteme identifiziert haben). Die Unternehmen sind sich dieser Tatsache bewusst und bemühen sich selbsterklärende Anwendungen einzusetzen.

Final kann bzgl. der Informationstechnologien festgehalten werden, dass ihre Realisierung im Hinblick auf Funktionalität und Implementierung suboptimal ist. Eine bessere Abstimmung auf die damit durchzuführenden Prozesse sowie die Anwender der Systeme scheint notwendig zu sein.

Mitarbeiter

Alltag

Die meisten Unternehmen bieten für ihren neuen Mitarbeiter *Trainingsprogramme* an, um den Einstieg in den Geschäftsalltag zu erleichtern. Dabei handelt es sich jedoch nicht um explizite Trainingseinheiten zum Thema (Geschäfts-) Prozessmanagement; diese scheinen auch (wie bereits von den anderen Studien im internationalen Umfeld festgestellt wurde, siehe Kapitel

2.3.2.1) im deutschen Firmenleben unterrepräsentiert zu sein. Trainings werden ansonsten nur zu speziellen technischen Systemen angeboten. Im eigentlichen Tagesgeschäft werden den neuen Mitarbeiter Mentoren zugewiesen, die sie beim Einleben unterstützen sollen. „*Training on the Job*“ ist eine favorisierte Methode, um neue Mitarbeiter zu integrieren.

Die übergreifende, ganzheitliche Einbindung der Mitarbeiter in den Geschäftsbetrieb erfolgt generell über die Projektorganisation, die typische Organisationsform bei Anwendung des Prozessmanagementansatzes (siehe zum Beispiel [BeKR05]). In den meisten Fällen, sieben von neun Unternehmen haben sich entsprechend geäußert, wird dabei im Team gearbeitet.

Zur Frage, was im Hinblick auf das Tagesgeschäft (und bzgl. des Umgangs mit den Mitarbeitern) verbessert werden könnte, kamen sehr unterschiedliche Antworten. Allgemein klang durch, dass die Kommunikation bzw. Abstimmung zwischen den Mitarbeitern zu optimieren ist und damit eng verbunden auch die Organisationsstrukturen. Es besteht Übereinstimmung darin, dass die Diskussion der Schnittstellen ein wichtiger Punkt ist, die funktionsorientierte Denk- und Arbeitsweise weniger ausgeprägt sein sollte, sowie auch [SIBC07] eine zu geringe Integration des Prozessgedanken identifiziert werden konnte. Die Unternehmen setzen sich also prinzipiell für das Prozessmanagement ein, sehen aber auch dass zwischen Idee, den ausgearbeiteten Konzepten und der Realität noch eine Lücke klafft.

Qualifikation

Zur Umsetzung des Prozessmanagement-Ansatzes brauchen nicht alle Mitarbeiter eine entsprechende Qualifikation auf diesem Themengebiet. Wichtig ist es, dass die Mitarbeiter ein Grundverständnis von den Abläufen ihres Unternehmens haben. Dies geht einher mit der Anforderung, wie sie bereits im Hinblick auf die Prozessverwendung bzw. –Ausführung, festgestellt wurden. Dort wurde gefordert, dass die Prozesse für eine erfolgreiche Ausführung „in den Köpfen der Mitarbeiter“ sein müssen; ebenso ist „*der Mensch mit seinen Fähigkeiten und gesundem Menschenverstand*“ (Nr.8, Tabelle 2-1) essentiell. Im Zusammenspiel mit Computer oder auch Roboter ist diese Komponente essentiell für den Alltagsbetrieb des Prozessmanagement-Ansatzes. Darüber hinaus ist es natürlich wichtig, dass die Mitarbeiter prinzipiell eine gute fachliche Qualifikation besitzen, um die in den Prozessmodellen spezifizierten Aufgaben auch inhaltlich und fachlich ausführen zu können.

Fazit

Zusammenfassend kann, wie bei den internationalen Studien auch, festgehalten werden, dass es sich beim Prozessmanagement um einen sowohl auf strategischer als auch operativer Ebene etablierten Managementansatz handelt. Die Interviewpartner konnten eine wesentliche Verbesserung ihrer Ergebnisse bestätigen, vor allem im Fall einer kompletten Reorganisation des Betriebs.

Es scheint, dass der Fokus sehr auf den internen Abläufen liegt, da eine Integration externer Partner in diesen Management-Ansatz nicht explizit erwähnt wird. Entsprechende Konzepte wie SCM (= Supply Chain Management) oder CRM (= Customer Relationship Management) (siehe [ScSe08]) wurden lediglich einmal erwähnt. Eine holistischere Umsetzung des Prozessmanagements [Hung06], bei der die externen Partner in diesen (Qualitäts-) Management-Ansatz mit eingebunden werden [ZaSi95], könnte noch weitere Verbesserungen mit sich bringen. Da bei den internationalen Studien dies bzgl. keine Angaben gemacht wurden, ist an dieser Stelle kein Vergleich möglich.

Informationstechnische Systeme sind nicht das Hauptwerkzeug zur Umsetzung des Prozessmanagement-Ansatzes. Die Systeme scheinen bzgl. ihrer Eigenschaften und Funktionen Qualitätsmängel zu haben, was vor allem die Modellierungswerkzeuge betrifft. Dies wird durch die Tatsache, dass individuelle Unternehmensaspekte eigentlich überhaupt nicht modelliert werden, unterstrichen; nur die üblichen Aspekte werden modelliert, was mit den Ergebnissen der internationalen Studien korrespondiert. Generell hat es sich herauskristallisiert, dass die IT-Systeme und die Prozesse nicht optimal aufeinander abgestimmt sind. So wird in der Praxis dem Faktor Mensch bei der Ausführung eine hohe Bedeutung beigemessen.

Im Hinblick auf die beiden Studien konnten somit übereinstimmend die Schwachpunkte im Bezug auf den Prozesslebenszyklus bzw. den Ansatz des Prozessmanagements identifiziert werden. Generell ist zu erkennen, dass die erzielten Ergebnisse bzw. verwendeten bzw. vorhandenen Systeme, wie es zum Beispiel vor allem bei den Modellierungswerkzeugen der Fall ist, sich nicht immer mit den gewünschten Anforderungen decken. Dies deutet darauf hin, dass genau dieser Stelle, nämlich der richtigen Definition und Umsetzung der Anforderungen, zu wenig Beachtung geschenkt wird.

2.3.3 Fazit bzgl. dem Stand in Forschung und Praxis

Das grundlegende Verständnis, was den Ansatz des Prozessmanagements betrifft, deckt sich bzgl. Theorie und Praxis. Als Instrument zur Organisation des Geschäftsbetriebes ist es auf strategischer wie auf operativer Ebene anerkannt. Doch gibt es Diskrepanzen bei der Umsetzung der theoretischen Konzepte in der Praxis, so dass die Ergebnisse bei letzterer nicht immer das liefern, was die Theorie vorgegeben hat. Die Diskrepanz gilt es jedoch differenziert zu betrachten. Es ist zu berücksichtigen, dass die in der Theorie bzw. im akademischen Bereich ausgearbeiteten Ideen und Konzepte unter anderem zukünftige Möglichkeiten darstellen und somit stets noch verbesserungswürdig sind; es handelt sich um Vorstellungen, die noch nicht (vollständig) in der Praxis erprobt wurden, wodurch die Feinabstimmung fehlt. Es ist zu vergleichen mit einem Prototyp aus der Produktentwicklung, der sich auch erst unter echten Bedingungen bewähren muss. Aufgabe der Praxis ist es, herauszufinden, was in welcher Form aus der Theorie übernommen werden kann; die Praxis ist zunächst der „Tester“ der akademischen Ideen. Die Diskrepanzen sind somit ein

essentieller Bestandteil für die Weiterentwicklung bzw. kontinuierliche Verbesserung der Konzepte. Abzugrenzen sind davon explizit Ideen und Konzepte, bei denen selbst nach längerer Zeit (das Thema Prozessmanagement wird insgesamt seit ungefähr 20 Jahre diskutiert) noch Diskrepanzen zwischen Theorie und Praxis zu finden sind. In solchen Fällen ist eine grundlegende Analyse des vorliegenden Falls unabdingbar.

Die hohe Relevanz der Prozessmodellierung kann mit den analysierten Studien sowie den Ergebnissen der eigenen klar aufgezeigt werden. Mittels Geschäftsprozessmodellen oder IT-Modellen werden notwendige Informationen für den Geschäftsbetrieb, organisatorische Strukturen wie auch die informationstechnische Aspekte in einer konsolidierten Art und Weise festgehalten [Giag01] [LiPr96]. Die Modelle bilden die Grundlage für die spätere Ausführung der Prozesse sei es mit oder ohne IT-Systeme.

Für eine erfolgreiche Realisierung der Modellierung wird in der Literatur die Designphase inklusive der dabei durchzuführenden Anforderungsanalyse (siehe dazu Kapitel 2.3.1) als ein essentieller Schritt betrachtet. In keiner der in Kapitel 2.3.2 betrachteten Studien (vor allem bei denen, die allgemein das Thema Prozessmanagement behandeln) sind jedoch explizit Informationen über die Designphase und methodische Vorgehensweisen, Rahmenbedingungen oder zu erreichenden Ziele zu finden. Selbiges gilt für die Anforderungsanalyse. Auch in der eigenen Studie sind nur sehr wenige Informationen dazu zu finden.

Das seit der Veröffentlichung des Artikels von Hevner [HMPR04] im Jahre 2004 gestiegene Interesse an diesem Thema [InRe08] scheint den Alltag der Unternehmen noch nicht erreicht zu haben, obwohl die jüngsten der analysierten Studien aus dem Jahre 2010 stammen, so dass ein gewisser Zeitrahmen zur Übernahme der Ideen inzwischen vorhanden ist. Das Design scheint in der Praxis eine weitaus geringere Relevanz als die sich anschließende Modellierung zu haben; und dies obwohl das Design die notwendigen Angaben für die Modellierung liefert. Die Prozessmodelle sind letztendlich „nur“ ein Instrument, um die im Design ermittelten Anforderungen und Umsetzungsmöglichkeiten festzuhalten [DaSh90]. Somit ist die unterschiedliche Relevanz von Design und Modellierung, wie sie identifiziert wurde, nicht nachvollziehbar, jedoch erklärbar: Es scheint noch eine starke Vermischung dieser beiden Phasen vorzuliegen, was sich vor allem darin zeigt, dass die Aufgaben des Designs weniger vorbereitend im Bezug auf die Modellierung, sondern wenn dann begleitend erfolgen. Zeit- und Kostendruck, die dazu führen existierende Lösungen (vor allem im Bezug auf Modellierungssprachen und -Werkzeugen) ohne ausreichende Analyse zu übernehmen, können ein Grund dafür sein. Auch scheint es noch an einer klaren Trennung von Verantwortlichkeiten in dem Sinne, dass „Designer“ ihrer Aufgabe nachgehen, zu fehlen.

Da die Designphase in der Praxis nur in geringen Ausmaß bzw. fast gar nicht durchgeführt wird, ist erklärbar, dass in den Prozessmodellen der befragten Unternehmen meist nur die gängigen Aspekte (wie Prozesse, Rollen und Werkzeuge [CuKO92] [JaBu96]) dargestellt

werden, die Modelle nicht zum jeweiligen Anwendungskontext passen und dem Anwender wichtige Informationen fehlen. Auch die IT-Systeme, die letztendlich auf Basis von Modellen entwickelt werden [VeCo94], können folglich nicht zu den durchzuführenden Prozessen passen, vor allem im Bezug auf Funktionalitäten. Es werden vorhandene (Modellierungs-) Werkzeuge, ohne oder mit nur geringen Anpassungen wieder bzw. weiterverwendet, ohne dabei auf den genauen Zweck und Kontext der Prozesse einzugehen und, speziell bei den Modellierungswerkzeugen, auf die „Richtigkeit“ der Modellierungssprachen zu achten [ClSW08]. Da die Designphase am Anfang des Prozesslebenszyklus steht, setzen sich Fehler in alle nachfolgenden Phasen durch und ziehen möglicherweise weitere mit sich. Zwar können sie gegebenenfalls mit geeigneten Maßnahmen ausgeglichen werden, doch nimmt dies zusätzlich Zeit und Kosten in Anspruch.

Da ein klares Defizit bzgl. der Beachtung bzw. Durchführung der Designphase festgestellt werden konnte, was bereits mit der unzureichenden Analyse der Anforderung beginnt, gilt es in dieser Arbeit aufgrund der Bedeutung der Designphase deren Aufgaben sowie Ziele zunächst explizit herauszuarbeiten. Die ermittelten Aspekte werden schließlich in einem methodischen Rahmenwerk für das Design integriert. Dies erfolgt im Hinblick auf den gesamten Prozesslebenszyklus, insbesondere der Modellierung.

Kapitel 3

Design und Modellierung – Definition und Grundlagen

Nachdem in dem vorherigen Kapitel die Begriffe Design und Modellierung bereits kurz eingeführt worden sind, erfolgt nun eine klare Abgrenzung dieser beiden Phasen. Dazu werden *Design* und *Modellierung* zunächst jeweils allgemein definiert, um sie dann auf den Kontext des Prozessmanagements zu übertragen. Für einen umfassenden Überblick werden die methodischen Vorgehensweisen der beiden Phasen erläutert.

3.1 Design und Modellierung – eine Abgrenzung

Von einer sauberen Spezifikation wird eine klare Trennung der beiden Phasen erwartet, so dass mit der eindeutigen Klärung der Aufgabenstellung auch die zu erzielenden Ergebnisse klar definiert sind. Eine eindeutige Zielvorstellung wirkt sich wiederum auf die Qualität der Ergebnisse der beiden Phasen aus.

3.1.1 Design

Der englische Begriff *Design* bedeutet ins Deutsche übersetzt „Entwurf“ [Weis91]. Er wird in den verschiedensten Sprachen verwendet, so zum Beispiel im Französischen oder Italienischen. Der deutsche Sprachgebrauch fokussiert sich eher auf den formalen bzw. künstlerischen Aspekt und verdinglicht den Designbegriff, während der angelsächsische Designbegriff auch den technischen sowie konstruktiven Anteil des Designs umfasst. Eine sprachübergreifend einheitliche Definition des Begriffes gibt es also nicht [RaWa09].

Design ist in vielen Bereichen ein zentrales Thema. Im Bereich der Architektur und Kunst ist der Begriff wohl am bekanntesten [WaWE92]. Er wird aber zum Beispiel auch im Ingenieurbereich, nämlich in der Konstruktion, verwendet [DyLi00] [PaBe97] [Petr06] und im Bereich der Informationssysteme. Im Bereich der Informationstechnik geht es um die Organisation eines Unternehmens (Strategie, Kultur und Prozesse) und die Gestaltung der Technologien (Infrastruktur, Anwendungen, ...) [HMPR04].

[Schn05] und [Stef00] definieren das Design als einen *Gestaltungs- und Entwurfsvorgang*, dem eine eingehende *Analysephase* bzgl. des zu gestaltenden Objektes vorausgeht. Letztendlich gilt es verschiedene Funktionen zu erfüllen. Einzelne Funktionen werden im Folgenden allgemein erläutert und es wird gezeigt, wie sie im Kontext Prozesslebenszyklus im Rahmen des (Geschäfts-) Prozessmanagements (siehe Kapitel 2.1) zu sehen sind.

Zeichenhaft-semantische Funktion

Das Design soll eine Idee, einen Begriff oder Vorgang symbolisch und somit wahrnehmbar bzw. verständlich darstellen. Mit welchem Symbol oder Zeichen eine Vorstellung eindeutig assoziiert werden kann, muss zuvor genau ermittelt werden, um Missverständnisse oder sogar ein Nichtverstehen zu vermeiden [Schn05] [Stef00]. So symbolisieren zum Beispiel üblicherweise Schlegel und Eisen den Bergbau. Bekannt ist auch der sog. Mercedes-Stern, der die entsprechende Automarke repräsentiert.

Auf das Prozessdesign bezogen geht es um die Festlegung der textuellen, tabellarischen und/oder graphischen Darstellungsarten (Zeichen, Symbole, Syntax) zur Präsentation der Abläufe einer Organisation [HMPR04] [JaDV08]. Es beinhaltet die Definition von Zeichen, Symbolen mit entsprechender Syntax (welche die Beziehung zwischen den Zeichen und Symbolen festlegt) und Semantik (welche die Bedeutung der Zeichen und Symbole definiert) [Bode06]. In ihrer Gesamtheit bilden diese drei die Definition einer (Prozess-) Modellierungssprache [Bode06] [CISW08].



Abbildung 3-1 Beispiel zeichenhaft-semantische Funktion

Eine mögliche Realisierung dieser Designfunktion ist in Abbildung 3-1 zu sehen. Rechtecke werden als Prozessschritte angesehen, die jeweils über einen Kontrollfluss in Form eines Pfeils miteinander verbunden, d.h. in eine Ablaufreihenfolge gebracht werden. Sollen auch Dokumente angegeben werden, müssen weitere Zeichen inklusive Semantik und Syntax definiert werden. Für eine weitere Optimierung kann zum Beispiel für das Prozessmodell einer Klinik festgelegt werden, dass die Modellierungssprache zwei Arten von Schritten beinhalten soll: ovale und eckige Schritte zur Unterscheidung administrativer und medizinischer Abläufe.

Formale Funktion

Die von außen sichtbare Gestalt eines Objektes oder der Ablauf einer Handlung (nicht der Inhalt) müssen bzw. sollen bestimmten Vorschriften genügen. Gestalt oder Handlung entsprechen dann einem nach außen sichtbaren Rahmenwerk, das der allgemeinen Orientierung sowie dem Verständnis dient. Diese Vorschriften gilt es im Design entsprechend auszuarbeiten [Stef00] [Schn05].

Dies entspricht im übertragenen Sinn der syntaktischen und semantischen Korrektheit der (Prozess-) Modellierungssprachen sowie der damit erstellten Modelle [FrvL03] [Mend08]. Dies bedeutet, dass die Elemente eines Prozessmodells gemäß den festgelegten Vorschriften,

sog. Modellierungsregeln, erstellt werden. So muss ein Prozessschritt immer mit mindestens einem ein- und ausgehenden Kontrollfluss oder einem Datenelement (bzw. Datenfluss) definiert werden, um syntaktisch korrekt zu sein. Die formalen Aspekte können sich auch aus dem Anwendungskontext der Prozesse ergeben, durch gesetzliche Vorgaben oder Unternehmensrichtlinien, die zum Beispiel die parallele Ausführung administrativer Schritte untersagen. Es zeigt sich hier, dass sich die Umsetzung der Anforderungen sowohl auf das Design als auch auf die Verwendung einer Modellierungssprache beziehen kann.

Ästhetische Funktion

Das Ergebnis des Gestaltungsvorgangs soll nicht nur seinen Zweck an sich erfüllen, sondern sollte auch ansprechend und akzeptabel sein [Schn05] [Stef00]. Was ansprechend bzw. akzeptabel ist richtet sich nach den Nutzern. Diese definieren die Anforderungen, ihre „Schönheitsideale“, die es dann funktional umzusetzen gilt. Die Problematik ist hierbei, dass Ästhetik nur schwer objektiv definiert und beurteilt werden kann.

Dies kann direkt auf die Modellierungssprache und ihre Benutzerfreundlichkeit sowie auf die Modelle bzw. Modellierung als Ganzes übertragen werden [FrvL03]. Die Modelle sollen letztendlich ansprechend und lesbar sein; es reicht nicht aus, sich auf die rein inhaltliche oder formale Darstellung zu konzentrieren. So werden bei EPKs (= Ereignisgesteuerte Prozessketten) Modellierungskonstrukte für Papierdokumente, elektronische Dokumente und allgemeine Informationsobjekte in ihrer graphischen Darstellung unterschieden [KeNS92]. Dennoch ist eine gewisse Ähnlichkeit und damit Zusammengehörigkeit zu erkennen, wie es Tabelle 3-1 zeigt. Lesbarkeit und Verständnis erhöhen sich dadurch.




	Graphische Darstellung	Anmerkung
Dokument in Papierform		Bsp. Brief
Elektronisches Dokument		Bsp. E-Akte
Informationsobjekt		Informationsobjekt als ein Element des Datenflusses

Tabelle 3-1 Beispiele Informationsobjekte EPK

Aus ästhetischen Gründen und weil ein Leser nur eine begrenzte Zahl an Elementen auf einmal erfassen kann [Mill56], sollten nicht mehr als sieben Prozesselemente auf einem Blatt modelliert sein. Diese Anforderung zeigt wiederum, dass sich eine Anforderung nicht direkt auf die Definition der Modellierungssprache beziehen muss, sondern auch deren Verwendung bei der Modellerstellung betreffen kann. Außerdem handelt es sich um eine Anforderung, die auch der formalen Funktion zugeordnet werden kann. Gesagtes gilt auch für die Einfärbung

der Datenelemente bei der Erstellung eines Modelles gemäß vorab in der Designphase definierter Kategorien; die medizinischen Daten werden demnach zum Beispiel rot, die administrativen blau eingefärbt.

Strukturierende Funktion

Die strukturierende Funktion dient der Verdeutlichung des Aufbaus eines zu gestaltenden Objektes bzw. der Gliederung von dessen Bestandteilen [PaBe97] [Schn05]. Die Zusammenhänge zwischen den Einzelteilen des Gesamtobjektes und deren Ordnung müssen daher zuvor grundlegend analysiert werden. Dabei kann zum Beispiel zwischen hierarchischen oder logischen Zusammenhängen [PaBe97] unterschieden werden.

Genauso müssen auch die Struktur und die Zusammenhänge des zu modellierenden Sachverhaltes in einem (Prozess-) Modell verdeutlicht werden. Die Prozesse lassen sich anhand verschiedener, voneinander unabhängiger Perspektiven (zum Beispiel funktions-, datenorientiert [JaBu96]) logisch strukturieren. Darauf aufbauend kann die Definition aspektorientierter Modellierungssprachen erfolgen [JaBu96]. Es sind Modellierungskonstrukte und Regeln zu definieren, wie zum Beispiel ein kompositer Prozess, der es erlaubt, mehrere logisch zusammengehörende Schritte zusammenzufassen und somit Zusammenhänge zu definieren.

Der strukturierende Aspekt geht oftmals mit der ästhetischen sowie zeichenhaft-semantischen Funktion einher. In dem bereits in Kapitel 2.2.3 erwähnten Prozessmodell wurde eine hierarchische Gliederung in Phasen, Hauptprozesse und Arbeitsschritte vorgenommen. Dies wurde zusätzlich ergänzt durch Verwendung verschiedener Farben für den funktionalen Aspekt. Letzteres dient zum einen der zusätzlichen Strukturierung, kann zugleich aber der ästhetischen Funktion zugeordnet werden. Denn die farbliche Kennzeichnung wirkt sich positiv auf die Lesbarkeit des Modells aus, da dem Anwender klar ist, auf welcher Ebene er sich befindet. Dieses Beispiel zeigt, dass die einzelnen Funktionen des Designs nicht isoliert voneinander zu betrachten sind, sondern oftmals ineinander greifen und sich gegenseitig ergänzen.

Didaktische Funktion

Schließlich möchte der Designer dem späteren Anwender mit seinem von ihm entwickelten Objekt auch immer etwas vermitteln, d.h. er verfolgt ein bestimmtes Ziel [Schn05]. Dabei kann es um die Vermittlung von Tatbeständen, von Wissen und/ oder einem bestimmten Kenntnissen gehen [Ball70]. Dies muss zunächst inhaltlich ermittelt werden, um dann zu definieren, wie es didaktisch vermittelt werden kann. Dies ist vor allem mit der ästhetischen Funktion eng verbunden.

Mit jedem Element der Modellierungssprache und dessen Verknüpfung zu weiteren Elementen soll ein Ausschnitt aus der Realität vermittelt werden. Die im Kontext dieser Arbeit erstellten Prozessmodelle sollen dem Nutzer einen Überblick darüber geben, welche

Aufgaben die jeweilige Organisation zu erfüllen hat und wie diese Tätigkeiten zum Beispiel rollen- und / oder abteilungsübergreifend miteinander zusammenhängen. Die didaktische Funktion bezieht sich somit zunächst auf das Modell und muss dann auf die Modellierungssprache herunter gebrochen werden. Für eine bessere didaktische Vermittlung eines Sachverhaltes kann dann zum Beispiel definiert werden, dass nie mehr als sieben Elemente auf einem Blatt gezeichnet werden [Mill56]. Dieser Aspekt wurde bereits bei der ästhetischen Funktion genannt, was zeigt, dass mit einem im Design definierten Aspekt gleich mehrere Funktionen erfüllt werden können.

Folgendes kann somit für das Prozessdesign festgehalten werden:

Zu Beginn der Designphase muss zunächst eine genaue *Anforderungsanalyse* im Hinblick auf den abzubildenden Sachverhalt, den Zweck der Abbildung und das dafür gewünschte Prozessmodell mit seinen individuellen Eigenschaften erfolgen (= *Input*).

Darauf aufbauend gilt es, die notwendigen *Funktionen* zu realisieren, was in Form der *Modellierungssprache* und deren Modellierungselementen, deren Eigenschaften und *Modellierungsvorschriften* erfolgt. Dies kann durch Anpassung oder Auswahl einer existierenden Modellierungssprache oder die komplette Neudefinition erfolgen. Dabei gilt es zu beachten, dass sich im Design eine Anforderung auf mehrere Funktionen beziehen kann bzw. umgekehrt eine Funktion auf mehrere Anforderungen. Desweiteren beschränkt sich die Umsetzung der Funktion nicht nur auf das Design, sondern erstreckt sich auch auf die nachfolgenden Phasen des Prozesslebenszyklus.

Das *Ergebnis* (= *Output*) ist zunächst eine *Modellierungssprache*, welche die diskutierten Funktionen im Hinblick auf den aktuell vorliegenden Kontext unterstützt bzw. erfüllt. Sie wird ergänzt durch die Angabe von *Modellierungsregel* zur Erstellung eines Modells. Dies führt letztendlich zur Festlegung der *konzeptionellen Modellgestaltung*, was explizit von der Modellierung eines konkreten Sachverhaltes abzugrenzen ist. Auch können Aspekte definiert werden, die sich auf die Verwendung des Models beziehen.

3.1.2 Modellierung

Ursprünglich wurde der Begriff *Modell* vor allem im Bereich der Architektur verwendet. Dort bezeichnet das italienische Wort „modello“, das aus dem Lateinischen (modellus) stammt (deutsch: Maßstab), einen Maßstab [Mitt84]. Der bezeichnete Maßstab ist Teil eines Messvorgangs, der einen Ausschnitt der Realität in eine für den Menschen verständliche, „leichter realisierbare“ [Mitt84] Darstellung umsetzt.

Auch heutzutage werden bei Gestaltungs- und Entwicklungsvorgängen zum besseren Verständnis und zum Zweck der Kommunikation Objekte oder ein Sachverhalt vereinfacht,

also nur mit den wesentlichen Aspekten, Strukturen und Verhaltensweisen, in Modellen festgehalten, die für unterschiedliche Zwecke verwendet werden. Drei Anwendungsbeispiele sind in Tabelle 3-2 zu sehen. Verschiedenste Modellierungssprachen und dazugehörige Modellierungswerkzeuge existieren zum Erstellen der jeweiligen Modelle [Andr94] [GeWa97] [MBKP04] [PaBe97] [Remu02].

Zur weiteren Konkretisierung der Abgrenzung von Design und Modellierung wird hier eine Einführung in den Modellbegriff und die Modellbildungstheorie gegeben, um diese dann auf den Kontext der Prozessmodelle zu übertragen. Die allgemeine Modelltheorie findet in der Wissenschaft eine breite Akzeptanz (vgl. auch [Schü98] und [Teub99] sowie die dort zitierte Literatur). Einen umfassenden Ansatz zur Modelltheorie liefert Stachowiak in seinem Werk [Stac73].

	Beschreibung
Modelle in der <i>Informatik</i>	Es erfolgt die Abbildung eines Realitätsausschnittes oder der Arbeitsweise eines (informatischen) Systems im Hinblick auf die informationstechnische Unterstützung bzw. Umsetzung [BrSt04] [RoPr00]. Beispiele für die Abbildung eines Realitätsausschnittes sind die in dieser Arbeit aus den Projekten des Lehrstuhls für Angewandte Informatik IV präsentierten Prozessmodelle.
Modelle im <i>Ingenieurwesen</i>	Komplexe Sachverhalte (sei es ein Produkt = technisches Erzeugnis oder ein Prozess) werden zum besseren Verständnis oder zur weiteren Bearbeitung in Modellform dargestellt [Ehrl07]. Es können mentale Modelle, Informationsmodelle sowie rechnerinterne Modelle unterschieden werden, mit unterschiedlichem Grad an formaler Abstraktion [PaBe97]. Beispiele sind in Abbildung 1-1 zu sehen.
Modelle in der <i>Wirtschaftswissenschaft</i>	Es erfolgt die Abbildung ökonomischer Strukturen und Prozesse zum Beschreiben, Erklären sowie zur Entscheidungsunterstützung [Bont04] [Gabl10]. Die wesentlichen Aspekte sind die der Aggregation (Zusammenfassung gleichartiger Individuen), der rationalen Verallgemeinerung (homo oeconomicus ersetzt den Mensch) und der Isolierung (ceteris paribus, das heißt unter sonst gleichen Bedingungen). Als Beispiel wären Kostenmodelle von Unternehmen zu nennen.

Tabelle 3-2 Beispiele für Verwendung von Modellen

Es werden nach [Stac73] durch ein Modell im Allgemeinen im Folgenden dargestellte Funktionen erfüllt, die sich gleichermaßen auf den Kontext der Prozessmodelle im Speziellen übertragen lassen. Dabei lassen sich auch Aspekte des Prozessdesigns mit einbinden, was die

enge Kopplung der beiden Phasen und die Bedeutung der Designphase noch einmal verdeutlicht:

Abbildende Funktion

Modelle dienen der Darstellung der Wirklichkeit, eines Originals [HeMa08]. Dabei referenzieren Modelle das Original, zu dem sie in einer Abbildungsrelation stehen [vBro03]. Es sind Abbildungen oder Repräsentationen von natürlichen oder künstlichen Originalen möglich, die wiederum Modelle sein können [Stac73].

Bei Prozessmodellen stellt das Original die Abläufe und Tätigkeiten einer Organisation und die dabei zu beachtenden Aspekte, wie zum Beispiel Rollen, Werkzeuge oder Dokumente dar. Die Rollen können dabei zusätzlich in einem Organisationsmodell, das die Organisationsstruktur definiert, abgelegt werden [Buss98] [Gabl10]. Die Dokumente bzw. Daten und ihre Beziehungen können in einem separaten Datenmodell abgebildet sein [Gabl10]. Es kann jedoch immer nur das abgebildet werden, was die verwendete Modellierungssprache in ihrer Spezifikation vorgesehen hat. Sind zum Beispiel keine Konstrukte (Zeichen inklusive Semantik und Syntax) oder andere Konzepte zur Abbildung von Rollen definiert, können Rollen auch nicht Bestandteil des Modells sein. Die Grundlagen zur Realisierung der abbildenden Funktion sind somit bereits in der Designphase zu legen - eine Tatsache, die einmal mehr die Wichtigkeit des Designs hervorhebt.

Vereinfachende Funktion

Modelle sollten stets vereinfachend bezüglich der eigentlichen Realität [FrPr97] sein. Das Modell kann somit nicht alle Eigenschaften bzw. Attribute des Originals aufweisen, sondern nur solche, die den jeweiligen Modellierern und/ oder Modellnutzern relevant erscheinen [HeMa08] [Stac73]. Es erfolgt eine Verkürzung, d.h. Abstraktion des Originals, eine Konzentration auf das Wesentliche, so dass nur Ausschnitte dargestellt werden [LiPr96] [vBro03]. Die vorgenommene Abstraktion eines Sachverhalts soll dabei nicht das Ausmaß der Komplexität an sich verringern, sondern eine verständlichere und leichtere Darstellung und Kommunikation eines komplexen Sachverhaltes ermöglichen, was zur Beherrschung der Komplexität beiträgt [FrPr97] [Remu02].

Eine Vereinfachung bzw. Komplexitätsbeherrschung ist auch bei der Prozessmodellierung das Ziel. Von dem aktuellen Sachverhalt werden nur die relevanten Informationen angegeben. Zum Beispiel werden die Aufgaben nicht bis auf die Ebene einer detaillierten Aufgabenbeschreibung spezifiziert, sondern lediglich mit ihrer Bezeichnung genannt. Komposite Prozesse ermöglichen es Detailinformationen auf einer tieferliegenden Abstraktionsebene darzustellen. Das zweite Beispiel zeigt wieder die Bedeutung des Designs, da bereits dort das Modellierungsskonstrukt eines kompositen Prozesses zu definieren ist.

Pragmatische Funktion

Unter bestimmten Bedingungen und bzgl. einer bestimmten Fragestellung soll das Modell das Original ersetzen. Es stehen die Fragen nach dem für „wen, wann, wozu“ im Vordergrund, die durch den jeweiligen Kontext anwendungsbezogen und somit pragmatisch bestimmt werden [vBro03]. Die Objekte werden daher nicht in ihrer Gesamtheit abgebildet. Vielmehr erfolgt immer eine Betrachtung aus einer oder mehreren ausgewählten Perspektiven. Somit weisen die Modelle eine „Zeitlichkeit“, „Intentionalität“ und „Subjektivität“ auf [vBro03].

Selbiges ist direkt auf den Inhalt der Prozessmodelle übertragbar. So ist es bei der Abbildung von Prozessen eines Wirtschaftsunternehmens zum Beispiel entscheidend zu wissen, wer welche Schritte auszuführen hat (funktionaler und organisatorischer Aspekt), während bei einem chemischen Analyseprozess eines Labors die Angabe der Messgeräte inklusive der Messdaten wichtig sind (funktionaler, operationaler und datenorientierter Aspekt). Die beiden Modelle bzw. die Modellierungssprachen in ihrer Definition oder Verwendung unterscheiden sich sowohl in den Aspekten selber, als auch in deren Anzahl. Hier zeigt sich, dass sich Anforderungen auf verschiedene Art und Weise umsetzen lassen. Der datenorientierte Aspekt kann im ersten Beispiel bei der Modellierung schlicht weggelassen werden oder aber bei der verwendeten Modellierungssprache gar nicht erst definiert sein.

Folgendes soll für die Modellierung von Prozessmodellen festgehalten werden:

Als Ergebnis (=Output) der Modellierung wird die vereinfachte, pragmatische Abbildung eines Realitätsausschnittes gesehen, welche die Funktionen nach [Stac73] erfüllt. Das im Design erstellte Konzept eines Modells ist mit konkretem Inhalt gefüllt.

Die Funktionen an ein Modell können nur realisiert werden, wenn die verwendete Modellierungssprache die dazu erforderliche und geeignete Funktionalität besitzt. Die Modellierungssprache muss neben den in 3.1.1 genannten Funktionen auch die Modellfunktionen realisieren.

3.1.3 Fazit

Die Prozesse und ihre geplante Art und Weise der Realisierung inklusive des dazugehörigen Anwendungskontextes stellen klar die Anforderungen an das Design, den Gestaltungsvorgang. Diese Anforderungen müssen zu Beginn des Designs (und nicht erst kurz vor oder bei der Modellierung) ermittelt werden. Im Rahmen des Designs wird definiert, wie diese umgesetzt werden. Die Anforderungen werden auf die Modellierungssprache mit den Modellierungselementen und deren Eigenschaften sowie den Modellierungsvorschriften abgebildet. Beim Design erfolgt also eine konzeptionelle Modellgestaltung, die dabei erarbeiteten Ergebnisse werden bei der Modellierung dazu verwendet einen konkreten Sachverhalt abzubilden.

Dies verdeutlicht noch einmal die Bedeutung der Auswahl bzw. Definition oder Anpassung einer Modellierungssprache sowie der zu Beginn der Designphase durchzuführenden Anforderungsanalyse. Das Design hat eine weitaus höhere Bedeutung, als ihm aktuell im Vergleich zur Modellierung (siehe zum Beispiel [DGRI06] oder [Rose06a]) zugestanden wird und stellt eine essentielle Voraussetzung für eine erfolgreiche Modellierung dar. Ziel muss es somit sein, das Design klar von der Modellierung zu trennen und methodisch aufzubereiten. Dabei werden die für das Design definierten Aufgaben explizit definiert und als vorbereitende Phase zeitlich eindeutig vor die Modellierung positioniert.

3.2 Methodische Grundlagen des Designs

Gemäß dem vielfach zitierten Artikel von Hevner [HMPR04] ist es das Ziel des Designs Artefakte zu schaffen. Genannt werden dabei Konstrukte, Methoden, Modelle und die Instanziierung [Denn97] [HMPR04] [Tsic97] [Scho93] [Simo96]. Die vier Arten von Artefakten sind nicht isoliert voneinander zu sehen, sondern bauen aufeinander auf. Oftmals führt erst ihre Gesamtheit, die mit der Instanziierung abschließt, zu einem vollständigen Gesamtkonzept.

In der Zuordnung aller vier Artefakte zur Designphase zeigt sich die fehlende Trennung des Designs von der Modellierung (und den weiteren nachfolgenden Phasen), wie sie in Kapitel 3.1.1 ausgearbeitet worden ist. So sind die Modelle, die zur Repräsentation bzw. Abstraktion eines Sachverhaltes dienen, nicht mehr dem Design zuzuordnen, sondern stellen vielmehr das Ergebnis der Modellierung dar. Auch erfolgt im Design keine Instanziierung. Diese ist der Ausführung zuzuordnen. Unter Berücksichtigung der in den vorherigen Kapitel 3.1.1 herausgearbeiteten Definition dürfen für das Design „nur“ die Konstrukte und Methoden als relevant erachtet werden (siehe Tabelle 3-3), während die übrigen Artefakte den nachfolgenden Phasen innerhalb des Prozesslebenszyklus zugeordnet werden.

	Beschreibung
Konstrukte	Sie definieren Vokabeln und Symbole zur Definition und Kommunikation eines Problems oder einer Lösung. Im Kontext dieser Arbeit sind dies die Elemente der Prozessmodellierungssprache.
Methoden	Diese stellen Algorithmen oder Vorgehensweisen dar, um Probleme lösen zu können. Im Kontext der Prozessmodellierung beschreibt eine Methode mit welchen Schritten das Prozessmodell dargestellt wird [JaBS99].

Tabelle 3-3 Design-Artefakte nach [HMPR04]

Es werden nun folgende zwei Richtungen des Designs und dazugehörige methodische Ansätze kurz näher beleuchtet, um dem Leser ein umfassendes Bild dieser Phase des Prozesslebenszyklus geben zu können:

- *pragmatisches Prozessdesign*
- *theoretisches Prozessdesign*

Dabei ist zu analysieren, inwiefern sie mit der in Kapitel 3.1.1. herausgearbeiteten Definition von Design, als explizit vor der Modellierung durchzuführende Phase, konform sind.

3.2.1 Pragmatisches Prozessdesign

Beim pragmatischen Prozessdesign handelt es sich um einen praxisorientierten Ansatz [HMPR04] [ReLi04] [ReNV07]. Mittels informeller Richtlinien und Heuristiken, die auf *Erfahrungswerten* basieren, werden die Prozesse gestaltet (siehe zum Beispiel [Barr05], [MCLP99] oder [RuRu94]). Unter Berücksichtigung der Unterteilung der Artefakte nach [HMPR04] liegt der Schwerpunkt dabei im methodischen Bereich. Eine Erstellung konkreter Modelle sollte gemäß der Definition des Designs aus Kapitel 3.1.1. noch nicht erfolgen. Grundlage des Ansatzes bilden bereits existierende Prozesse bzw. Modelle, die durch Anwendung einer geeigneten Methodik verbessert werden sollen [AlGa99]. Ziel ist es, Schwachstellen zu eliminieren oder zumindest zu minimieren [LiRe05] [ReNV07]. Dies erfolgt meist durch lokale, d.h. begrenzte Änderungen an einem Prozessschritt oder jeglichem anderen prozessbezogenen Aspekt. Somit kann dieser Ansatz auch dem sog. *Re-Design* zugeordnet werden, wie es zum Beispiel in [DaSh90] beschrieben ist [HaCh03] [ZaSi95].

Im Laufe der Zeit haben sich die folgenden Ansätze etabliert:

- *Best Practice und*
- *Benchmarking*

Best Practice

In den letzten 20 bis 30 Jahren haben sich in den verschiedensten Anwendungsbereichen des Prozessdesigns bzw. des Prozessmanagements zahlreiche Best Practices entwickelt (siehe [Atef97] [DaSh90] [HaCh93] [HMPR04] [LiRe05] [ReLi04] [ReNV07]). Beispielhaft zu nennen wären das Gesundheitswesen, die Fertigungswirtschaft oder die Softwareentwicklung (zum Beispiel [Butl96], [Golo97] oder [OoGC11]).

Wörtlich übersetzt bedeutet der Begriff *Best Practice* „bestes Verfahren“ oder etwas freier übersetzt „Erfolgsrezept“ [Dict10]. Es handelt sich um in der Praxis bewährte Vorgehensweisen für bestimmte Arbeitsgebiete, um historische und geprüfte bzw. bestätigte Lösungen für ein bestimmtes Problem, die meist in Katalogen oder Handbüchern zusammengefasst werden [MCLP99]. Best Practices haben keinen individuellen Charakter, sondern sie werden generisch in den unterschiedlichsten Anwendungsfällen eingesetzt. Dazu

müssen sie fachkundig an die beim Anwendungsfall vorliegenden individuellen Bedingungen angepasst werden [Butl96].

Die Anpassungen können die verschiedensten Aspekte des Prozesses betreffen [LiRe05]: die Prozessteilnehmer sowie die Prozessausführenden oder die internen und externen Kunden, die einzelnen Prozessschritte, den Kontroll- bzw. Datenfluss, die Informationstechnologien bzw. die Werkzeuge und Systeme sowie die im Prozess zu verarbeitenden bzw. zu generierenden Informationen.

Zum besseren Verständnis werden im Folgenden exemplarisch einige Best Practices im Bezug auf den funktionalen (und verhaltensorientierten) Aspekt vorgestellt. Beispiele für Best Practices, die sich auf weitere Aspekte der Prozesse beziehen, sind in [ReLi04] zu finden. Weitere Aspekte zur Kontrollflussdefinition sind in [Böhm00] zu finden.

Best Practices im Bezug auf Prozessschritte - Bildung kompositer Schritte



Abbildung 3-2 Bildung kompositer Schritte aus [ReLi04]

Eine der meist zitierten Best-Practice ist die Bildung kompositer Schritte [ReLi04]. Dabei werden zur verbesserten Strukturierung der Prozesse mehrere „kleine“ Schritte zu einem übergeordneten Schritt zusammengefasst. Es gilt zu beachten, dass die Flexibilität der Durchführung sinkt, da die Schritte damit als eine zusammengehörende Einheit gesehen werden. Das kann die Qualität der Ergebnisse beeinträchtigen, sollte der composite Prozess zu umfassend werden und bei der späteren Ausführung nicht mehr zu bewältigen sein.

Umgekehrt kann ein „großer“ (meist sehr komplexer) Schritt in mehrere „kleine“, überschaubare Teilschritte unterteilt werden. Dies führt mit relativ hoher Wahrscheinlichkeit zu einer Erhöhung der Ergebnisqualität, da die Durchführung strukturierter ist. Aufgrund der höheren Anzahl an Schritten hat diese Vorgehensweise aber auch einen erhöhten Aufwand vor allem bei der Instanziierung zur Folge.

Best Practices im Bezug auf den Kontrollfluss – Resequencing

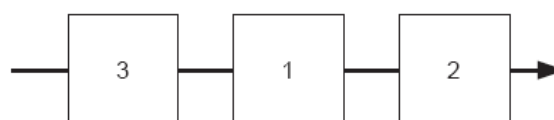


Abbildung 3-3 Resequencing aus [ReLi04]

Um Zeit- und Kostenersparnisse zu erzielen, bietet es sich oftmals an, die Reihenfolge der Schritte zu ändern, wodurch zum Beispiel unnötige Wartezeiten bei zwei aufeinanderfolgenden Schritten vermieden werden können [ReLi04]. Die Neuordnung in Abbildung 3-3 ist jedoch nur möglich, wenn Prozess 3 nicht darauf angewiesen ist, dass Prozess 1 und 2 vor ihm ausgeführt sind. Als eine Sonderform ist die parallele Anordnung von Schritten zu nennen [ReLi04] [Schu06], die vor allem eine Verkürzung der Durchlaufzeit zum Ziel hat.

Benchmarking

Das *Benchmarking* (= Maßstäbe setzen) basiert auf einem systematischen Vergleich der eigenen Organisation mit einer Vergleichsorganisation an Hand von Ähnlichkeitswerten bzw. Kennzahlen [Gabl10] [ZdKa02]. Es werden Produkte und Dienstleistungen, aber auch Prozesse betrachtet, d.h. strukturiert verglichen [Gabl10]. Letzteres ist auch unter dem Begriff *Process Benchmarking* geläufig [MCLP99] [Zair94] [ZaSi95].

Die Methode ist nur anwendbar, wenn zu den eigenen definierten Prozessen Vergleichsprozesse vorhanden sind. Die zu vergleichenden Prozesse müssen dabei derart definiert sein, dass ein Vergleich möglich ist.

Die vergleichende Analyse kann aus betriebswirtschaftlicher wie auch aus IT-bezogener Sichtweise erfolgen, um unter wirtschaftlichen Gesichtspunkten in qualitativer wie auch quantitativer Hinsicht einen Vergleich aufzustellen [MCLP99] [ZdKa02]. Im Vordergrund steht der Leistungsvergleich der Unternehmen bzw. hier der Unternehmensprozesse [ZaSi95] [ZdKa02]. Dazu werden unter anderem Kundenbefragungen oder Leistungsvergleiche auf Basis von veröffentlichten Statistiken durchgeführt [ZaSi95]. Die Analyse erfolgt anhand zuvor festgelegter Referenzwerte, wie zum Beispiel Durchlaufzeit, Absatzzahlen oder Kostenaspekte [MCLP99] [Zair94].

Werden relevante Abweichungen zwischen Referenz- und Ist-Wert identifiziert, gilt es entsprechende Änderungsmaßnahmen vorzunehmen, um die Abweichungen zu beheben oder zumindest zu verringern [Gabl10] [ZdKa02]. Die Relevanz der Abweichungen wird durch deren Ausmaß zum vorgegebenen Referenzwert definiert. Die Spezifikation individueller Referenzwerte ist notwendig, weil sich Unterschiede zwischen den Unternehmen und ihren Prozessen auch aus den individuellen Rahmenbedingungen ergeben können, die sich dann weniger zum Vergleich anbieten.

Das Benchmarking ist auch in engem Zusammenhang mit Best Practices (siehe Absatz oben) zu sehen. Denn das Analyseverfahren im Rahmen des prozessorientierten Benchmarking wird auch dazu verwendet Best Practices zu identifizieren [ZaSi95]. In diesem Zusammenhang wird auch von *Best Practice Benchmarking* gesprochen.

Bewertung des Pragmatischen Prozessdesigns

Leider führen die Methoden des pragmatischen Prozessdesigns nicht zwingend zu dem optimalen Design der Prozesse [ReNV07]. Dies resultiert daraus, dass die Methodik auf Erfahrungswerten aufbaut, die sehr von der persönlichen Wahrnehmung abhängen. Die Erfahrungswerte sind jedoch auch positiv zu bewerten. Bei damit gestalteten Prozessabläufen besteht ein gewisses Maß an Sicherheit, dass ein zufriedenstellendes und verwendbares Ergebnis erzielt werden kann. Aufgrund der Erfahrung können gewisse „Anfänger“-Fehler vermieden werden. Sowohl Best Practices als auch Benchmarking bieten den Vorteil, dass ein Vergleich mit bei anderen Unternehmen erfolgreich eingesetzten Lösungen erfolgt und die direkte Orientierung am Markt möglich ist.

Best Practices sind nicht ohne einige zusätzliche Arbeitsschritte einsetzbar, sondern müssen noch an die konkreten Bedingungen innerhalb der betreffenden Organisation angepasst werden. Dieser Anpassungsprozess sollte aufwandstechnisch nicht unterschätzt werden. Selbiges gilt auch für die im Rahmen eines Benchmarkings identifizierten Verbesserungsmaßnahmen. Nicht jedem Unternehmen ist dies aus organisatorischen oder finanziellen Gründen möglich. Beim Benchmarking ist zudem vor allem die Spezifikation der Referenzwerte ein kritischer Schritt. Das Unternehmen muss sich darüber im Klaren sein, welche Zielvorstellungen es genau verfolgt. Bei einem Vergleich mit anderen Organisationen, wie er beim Benchmarking vollzogen wird, besteht zudem stets die Gefahr, dass „Äpfel mit Birnen“ verglichen werden, auch wenn es sich um dieselben Kennzahlen jedoch in einer anderen organisatorischen Umgebung (zum Beispiel im Hinblick auf die Unternehmensgröße, den geographisch zu erreichenden Absatzmarkt) handelt. Es werden möglicherweise falsche Schlüsse gezogen, durch die unnötige, wenn nicht sogar kontraproduktive Maßnahmen veranlasst werden.

Bei den hier vorgestellten Methoden wird das Design bereits im Rahmen eines sich wiederholenden, d.h. verbessernden Prozesslebenszyklus durchgeführt. Von einer kompletten Neudefinition einer Modellierungssprache wird daher üblicherweise Abstand genommen, da dies eine sehr umfassende und aufwendige Neumodellierung des gegebenen Sachverhaltes zur Folge haben kann. Das Design beschränkt sich auf die Definition von Modellierungsregeln. Nach Gesagtem macht dies im Kontext eines kontinuierlichen Verbesserungsprozesses durchaus auch Sinn, so dass dies hier nicht negativ ausgelegt werden sollte.

Die pragmatischen Ansätze des Prozessdesigns bilden die Grundlage für die Entwicklung der im Folgenden beschriebenen theoretischen, wissenschaftlichen Ansätze. Die beiden Ausrichtungen des Designs ergänzen sich somit gegenseitig bzw. bauen aufeinander auf.

3.2.2 Theoretisches Prozessdesign

Im Gegensatz zum pragmatischen Prozessdesign handelt es sich bei der theoretischen Ausrichtung um einen formalen Ansatz, der auch unter dem Begriff *scientific design* geläufig

ist [HoVe01] [ReNV07] [vAHW03] [vAal98]. Es basiert auf formalen Methoden und Techniken und ist von einer sehr analytischen Vorgehensweise geprägt [HoVe01] [RLvA03]. Als *design from scratch* [ReNV07] [RLvA03] werden Prozesse grundlegend, ohne bereits existierende Prozesse zu berücksichtigen, neu gestaltet, weshalb auch von einem *revolutionären* Ansatz gesprochen wird [ReNV07] [RLvA03]). Dabei ist es das Ziel, den optimalen Prozessablauf und all seine zusätzlichen Eigenschaften zu spezifizieren.

Im Folgenden sollen exemplarisch Ansätze des theoretischen Prozessdesigns vorgestellt werden. Dies sind:

- *Product Based Design*
- *Algorithmen*

Product Based Design (PBD)

Der Ansatz des *Product Based Design* [ReNV07] [RLvA03] [vHRe00] ist im Kontext dieser Arbeit von Interesse, da er im Hinblick auf die beispielhaft verwendete Anwendungsdomäne der Produktentwicklung (siehe Kapitel 2.2.1) ein sehr naheliegender und nachvollziehbarer Ansatz ist. Hier bildet das Produkt den primären Ansatzpunkt zur Analyse, um die für die Fertigung des Produktes notwendigen Prozesse definieren zu können.

Das Product Based Design ist grundsätzlich eine Übertragung der Idee der Stückliste [Orli72] aus der Produktion- bzw. Fertigungswirtschaft auf den Bereich administrativer Prozesse [ReNV07] [vHRe00]. Es gilt die Annahme, dass das Produkt alle für den Herstellungsprozess notwendigen Informationen liefert. Es erfolgt daher zunächst die Identifikation der Produktbestandteile (= data elements) und deren logischen Abhängigkeiten (= production rules). Die sich daraus ergebende Struktur wird im sog. Produktdatenmodell (= PDM) festgehalten. In Abbildung 3-4⁴ ist beispielhaft ein solches Produktdatenmodell zu sehen. Die Knoten stellen die Bestandteile (= data elements) dar; mittels der Kanten werden die Abhängigkeiten verdeutlicht. Der Wurzelknoten an der obersten Stelle in der Abbildung stellt das Endprodukt dar.

Ausgehend von dem Produktdatenmodell kann darauf aufbauend das Prozessmodell erstellt werden. Folgende Methodik ist dazu definiert:

- Ein oder mehrere data elements werden zu einem *Herstellungsschritt* (=Activity) zusammengefasst. In Abbildung 3-4 werden zum Beispiel die neun data elements im linken unteren Teil des Produktdatenmodells der Activity G zugeordnet. Dieses Vorgehen wird für alle restlichen Knoten wiederholt, bis kein Knoten mehr übrig ist.
- Es erfolgt die Festlegung einer *Reihenfolge* der Herstellungsschritte (=Activity)

⁴ Die Bezeichnung der Knoten in der Abbildung ist in diesem Kontext nicht von Relevanz.

Mit diesen Informationen kann schließlich das *Prozessmodell* definiert werden. Damit erfolgt nach der Definition von Design und Modellierung in den Kapitel 3.1.1 und 3.1.2 der Wechsel zur Modellierungsphase, was in der hier zitierten Literatur nicht derart differenziert betrachtet wird.

Es handelt sich prinzipiell um eine sehr rationale und auch effiziente Vorgehensweise, da das Produkt als Grundlage des Designs verwendet wird. Jeder Schritt im Prozessmodell kann mit einem data element aus dem Produktdatenmodell begründet werden kann, so dass keine unnötigen Schritte im Prozessmodell zu finden sind. Die Vorgehensweise kann jedoch nicht verhindern, dass dennoch „schlechte“ Abläufe (zum Beispiel durch hohe Wartezeiten) definiert werden.

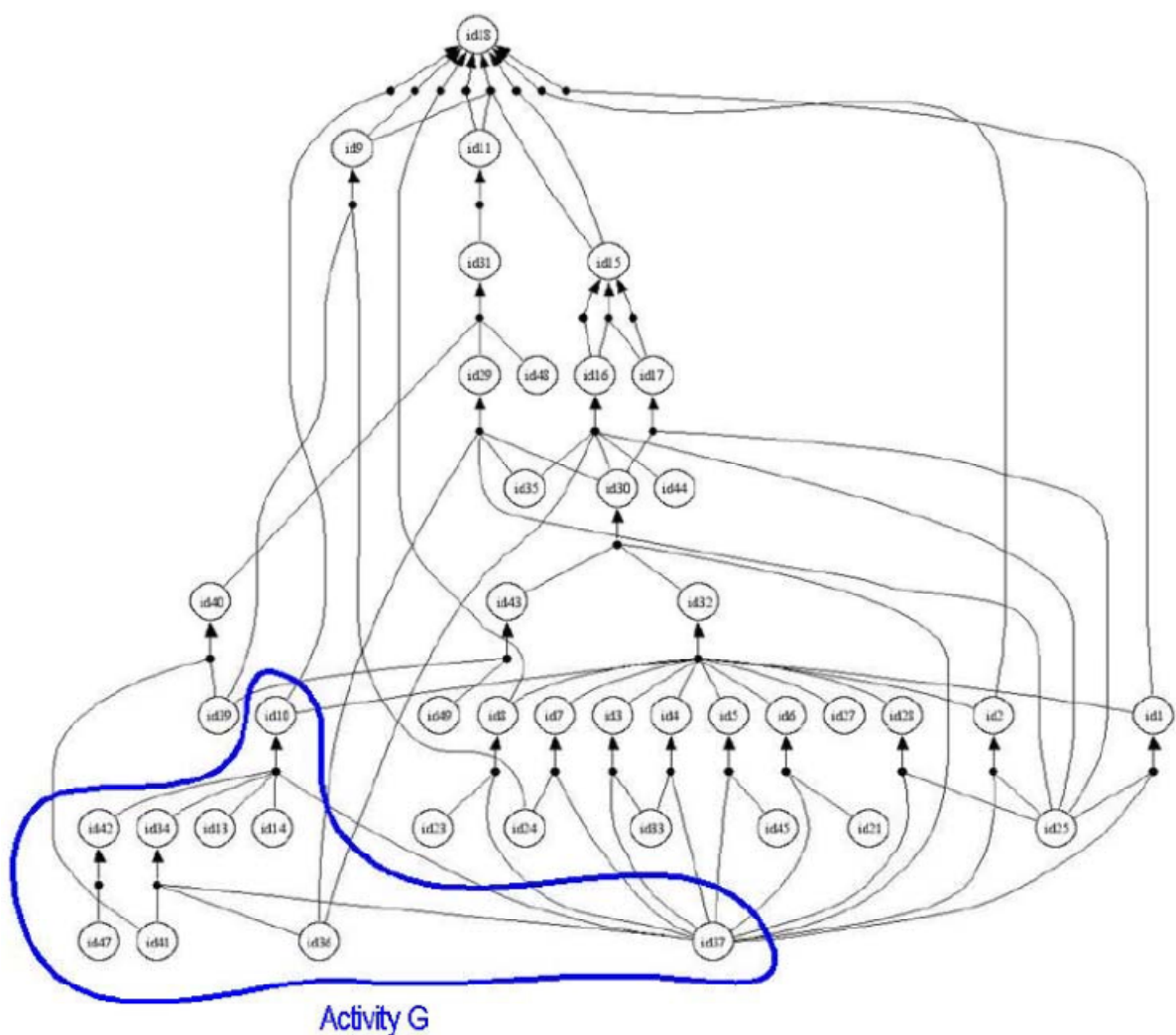


Abbildung 3-4 Produktdatenmodell aus [ReNV07]

Algorithmen

Algorithmen zeichnen sich dadurch aus, dass sie eine eindeutige Handlungsvorschrift zur Lösung eines Problems vorgeben [Levi02] [HoVe01] [VTMR07]. Dazu muss das Designproblem zunächst formal in einem Modell beschrieben werden, um dann unter Berücksichtigung von Optimierungsfunktionen (zum Beispiel additive oder Min-Max-Funktion) des optimale Design ableiten zu können. Dies erfolgt vornehmlich in Bezug auf die Aspekte Zeit und Kosten.

Auch im Bereich des Prozessdesign werden inzwischen Algorithmen angewendet, auch wenn dies aufgrund des formalen Charakters der Algorithmen gegenüber den oft nur qualitativ zu beurteilenden Prozessen schwierig ist [YuLi00] [VTMR07]. Im Folgenden werden drei Techniken, die auch im Rahmen des Prozessdesigns angewendet werden, vorgestellt. Die Erläuterungen sind nur kurz, da es sich um sehr mathematische Verfahren handelt - ein Themengebiet, das im Rahmen dieser Arbeit nicht im Fokus der Betrachtung steht.

- Im Rahmen der *Mathematischen Programmierung* [HoVe01] wird das Optimierungsproblem, hier der Prozess, in Form einer entsprechenden (Ziel-) Funktion erfasst inklusive der Angabe von Variablen und Constraints im Hinblick auf die Ressourcen, wie zum Beispiel Rollen oder Systeme. Berechnet werden damit zulässige und optimale Lösungen für das Problem. Es eignet sich jedoch eher nur für kleinere Fälle, vor allem was die Anzahl der Variablen betrifft.
- Bei größeren Design- bzw. Optimierungsproblemen kann die sog. *Branch-and-bound-Methode* [AVTa89] [HoVe01] [IcSZ93] verwendet werden. Im Vergleich zur mathematischen Programmierung ist sie leichter zu implementieren, hat meist eine kürzere Rechenzeit und liefert bessere Ergebnisse. Ziel der Methode ist es einen Entscheidungsbaum zu definieren, dessen Knoten Prozessschritte und dessen Kanten den Kontrollfluss darstellen. Beginnend mit der Auswahl einer Start-Aktivität wird dem entstehenden Baum jeweils eine weitere Aktivität hinzugefügt. Die Auswahl erfolgt unter Verwendung einer Bewertungsfunktion, die misst, ob der ausgewählte Knoten einen weiteren Beitrag zur Erzeugung des finalen Prozessoutputs liefert. Dieser Vorgang wird als *Branch-Schritt* bezeichnet und wird so oft durchgeführt bis das Hinzufügen eines weiteren Prozessschrittes zu keiner besseren Lösung führt. Der *Bound-Schritt* hat unterdessen die Aufgabe, bestimmte Zweige des Baumes mittels der Definition von Schranken "abzuschneiden". Dies bedeutet, dass diese in der weiteren Berechnung nicht mehr betrachten werden, um den Rechenaufwand zu begrenzen. Die Schranken werden auf Basis von Heuristiken festgelegt.
- Des Weiteren gibt es *genetische Algorithmen* [HoVe01] [LeMR94] [RuLe95], die auch nicht analytisch lösbare Probleme behandeln können. Sie generieren wiederholt verschiedene „Lösungsvorschläge“ für die Prozessmodelle. Dies setzt voraus, dass zuvor alle möglichen Prozesselemente definiert wurden. Die bereits erstellten Lösungen werden immer wieder verändert, miteinander kombiniert und kontinuierlich

einer Auslese unterzogen, so dass sich die Lösungsvorschläge den gestellten Anforderungen schrittweise annähern.

Wie bereits beim Product Based Design findet auch bei den hier vorgestellten Algorithmen eine Vermischung von Design und Modellierung statt. Während die Definition der Algorithmen dem Design zuzuordnen ist, sind die Ergebnisse der Auflösung bzw. Anwendung als Modelle zu verstehen. Die Auflösung der Algorithmen sowie die Ergebnisse sind somit der Modellierungsphase zuzuordnen.

Es gilt anzumerken, dass die Algorithmen in Bezug auf das Prozessdesign noch „in den Kinderschuhen“ stecken. So müssen sie bzgl. diverser prozessspezifischer Aspekte noch weiter entwickelt werden, wie zum Beispiel bzgl. der Berücksichtigung der Tatsache, dass ein Prozessschritt je nach Kontext unterschiedliche Outputs haben kann.

Bewertung des Theoretischen Prozessdesigns

Es handelt sich beim theoretischen Prozessdesign um eine sehr strukturierte und analytische Vorgehensweise, die zu eindeutigen und logisch nachvollziehbaren Ergebnissen führt. Fehler aufgrund von Mehrdeutigkeit einer Designlösung können vermieden werden.

Üblicherweise sind die Methoden zuvor in anderen Bereichen außerhalb der Prozessdomäne bei ähnlichen Problemstellungen erfolgreich angewendet worden (so zum Beispiel im Bereich des Ingenieurwesens oder der Fertigung), ehe sie auch im Bereich des Prozessdesigns Einzug gehalten haben. Sie basieren somit auf erprobten Verfahren, die bereits mehrfach optimiert wurden; sie gehen gewissermaßen aus Best Practices hervor und beruhen auf Erfahrungswerten bzw. sind historisch validiert. Daher sind vor allem die Verfahren, aber auch die Ergebnisse oftmals stabiler bzw. verlässlicher als Methoden des pragmatischen Prozessdesigns. Zudem lassen sie sich jederzeit in Bezug auf einen neuen Anwendungsfall wiederverwenden.

Doch muss berücksichtigt werden, dass die Umgebung, in der die Design-Artefakte schließlich angewendet, d.h. implementiert werden, aufgrund ihrer Unwägbarkeiten und fehlenden Planbarkeit sowie der kreativen Komponente der Prozessorientierung insgesamt nur schwer zu formalisieren ist [MaSc97] [HaCh93]. Die Ansätze sind aufgrund der Grundlagen und deren Historie gut aufgearbeitet und durchdacht, jedoch wird oftmals kritisiert, dass sie nur mit viel Aufwand umzusetzen und somit unpraktikabel sind [Reij03] [RLvA03]. Dem kann entgegen gehalten werden, dass zahlreiche Anwendungsfälle bekannt sind. Beispielhafte Berichte sind in [Reij03] und [RLvA03] zu finden. Es kann schwierig sein, die zu designenden Elemente und die dazugehörigen Kontextbedingungen so formal zu charakterisieren, wie es die theoretischen Methoden verlangen. Die Qualität eines Prozesses setzt sich zudem aus einer Vielzahl von Kriterien zusammen, die oftmals nur in Form eines Kompromisses zu erfüllen sind. Welcher Kompromiss der Beste ist, ist meist nur subjektiv, nicht jedoch formal zu beurteilen.

Der Hauptkritikpunkt an den Methoden des theoretischen Prozessdesigns besteht darin, dass keine klaren Grenzen zwischen Design und Modellierung getroffen werden. Die Ergebnisse des Designs werden nicht explizit bzw. separat als Konzept festgehalten, sondern werden vielmehr gleich in die eigentlichen Prozessmodelle integriert. Das Design bzgl. einer Modellierungssprache tritt somit in den Hintergrund. Die Strukturiertheit und Übersichtlichkeit der Vorgehensweise leidet darunter, dass sofort die Vermischung mit/ die Anwendung auf einen konkreten Sachverhalt erfolgt.

3.2.3 Fazit zu den Grundlagen des Designs

In Tabelle 3-4 ist eine Gegenüberstellung des pragmatischen und theoretischen Prozessdesigns an Hand ausgewählter Kriterien zu sehen. Die Auswahl der Kriterien hat sich aus der Analyse der beiden Ansätze ergeben.

Die Ansätze unterscheiden sich grundlegend in ihrem Ausgangspunkt, wodurch sich auch unterschiedliche *Einsatzzeitpunkte* im Verlauf des gesamten Prozesslebenszyklus ergeben. Beim pragmatischen Prozessdesign wird an existierenden Prozessen angesetzt, die an den identifizierten Schwachstellen überarbeitet werden. Demgegenüber startet der theoretische Ansatz „bei null“, weil entweder noch keine Prozesse definiert wurden oder der neue Prozess soweit wie möglich unvoreingenommen bzgl. bereits existierender Prozesse gestaltet werden soll.

Die Unterscheidung in *reaktives und aktives Vorgehen* ist mit dem eben Gesagten bereits implizit gegeben. Eine Reaktion kann sich immer nur an etwas Vergangenen bzw. Vorhandenem orientieren, wie es beim pragmatischen Ansatz mit den existierenden Prozessen als Ausgangspunkt der Fall ist. Ausgelöst wird eine Reaktion stets durch einen sog. „Reiz“, was den wesentlichen Unterschied zu einer Aktion ausmacht [BiIn09]. Eine Aktion, das Handeln an sich, findet zudem meist innerhalb eines vorgegebenen Regelwerks nach einem bestimmten Ablaufplan statt [Funk03], was mit den Erläuterungen zum theoretischen Prozessdesign einhergeht.

Formalisierung bedeutet die „Generalisierung einer (wissenschaftlichen) Aussage unter Absehung ihrer konkret-empirischen Bezüge“ [DeTV95]. Der *Formalisierungsgrad* der theoretischen Ansätze ist sehr hoch, während die pragmatischen Verfahren sehr stark auf Erfahrungswerten und somit empirischen Daten beruhen. Bei den Best Practices ist dies wohl am besten nachzuvollziehen.

Der pragmatische Ansatz geht von existierenden Prozessen aus. Diese werden in den meisten Fällen nur an ausgewählten Stellen, d.h. lokal, überarbeitet. Beim theoretischen Prozessdesign muss der gesamte Prozess vom Grunde auf gestaltet werden, da der Anspruch besteht, auf einem „weißen Blatt Papier“ zu beginnen. Das Design kann dann als global bezeichnet werden.

Im Hinblick auf den *Aufwand* ist festzustellen, dass beide Methoden, in ihrer Gesamtheit betrachtet, mit hohem Aufwand verbunden sind.

Während beim theoretischen Prozessdesign als *Ergebnis* ein optimales Prozessdesign das Ziel ist, ist es beim pragmatischen Prozessdesign „nur“ ein verbessertes Design. Diese Formulierung darf jedoch auf keinen Fall missverstanden werden. Sie bedeutet nicht, dass die Qualität bei Letzterem schlechter zu bewerten ist. Verbessert bedeutet hier gewissermaßen „pareto-optimal“, d.h. ein besserer Zustand bzgl. eines bestimmten Aspektes ist nicht möglich, ohne zugleich einen anderen Aspekt schlechter zu stellen [Bont04].

Ob letztendlich das pragmatische oder das theoretische Design besser ist, wird in der Literatur vielfach diskutiert (siehe dazu zum Beispiel [OnSo99]). Beide Ansätze haben ihren jeweiligen Einsatzzweck bzw. – zeitpunkt und jeweils ihren Berechtigungsgrund. Sie ergänzen sich gegenseitig und stehen in Wechselwirkung zueinander, wie es zum Beispiel auch [ReNV07] vertreten. Es muss also gemäß dem individuellen Kontext entschieden werden, welche Ausrichtung gewählt wird.

	pragmatisch	theoretisch
Ausgangspunkt	existierende Prozesse	„weißes Blatt Papier“
Einsatzzeitpunkt bzgl. Prozesslebenszyklus	nach den ersten Umläufen im Prozesslebenszyklus	vor der ersten Instanziierung des Prozesses oder nach mehreren Prozesslebenszyklusdurchläufen
Vorgehen	reaktiv	aktiv
Formalisierungsgrad	gering	hoch
Scope/ Umfang des Anwendungsbereichs	lokal: Optimierung einzelner Stellen des Gesamtprozesses	global: Gestaltung des Gesamtprozesses
Aufwand	hoch	hoch
Angestrebte Ergebnisqualität	verbessertes Prozessdesign	optimales Prozessdesign

Tabelle 3-4 Gegenüberstellung pragmatisches und theoretisches Prozessdesign

Es ist festzuhalten, dass bei den vorgestellten Ansätzen mit dem Ergebnis des Prozessdesigns meist schon die wirklichen Prozessmodelle betrachtet werden, die einen realen Sachverhalt abbilden. Dies widerspricht den Definitionen aus Kapitel 3.1.1 und 3.1.3 bzgl. Design und Modellierung, die eine Trennung von Design und Modellierung vorsehen, und ist daher als ein elementare Kritikpunkt zu betrachten. Das Ergebnis des Designs ist die rein

konzeptionelle Modellgestaltung, auf welcher der eigentliche Modellierungsvorgang dann aufbauen kann.

Mit dieser Kritik geht einher, dass sich beide Design-Ansätze darauf beschränken, die Prozesselemente (pareto-) optimal in einem Prozessmodell anzuordnen. Damit bewegen sich die Verantwortlichen in dem von den Modellierungssprachen *gegebenen Handlungsspielraum*, der sich auf die reine Verwendung der bereits vorhandenen Aspekte, Konstrukte oder Methodiken beschränkt, während eine Neudefinition nicht vorgenommen wird. Auch die verschiedenen Studien haben dieses Verhalten aufgezeigt (siehe Kapitel 2.3.2). Somit unterstreichen die Beschreibungen der verschiedenen Designmethoden die bereits geäußerte Kritik, dass das Design nicht in seinem vollen Maße ausgenutzt wird, was angesichts der Vermischung von Design und Modellierung auch nur sehr schwer möglich ist. Um die Problemstellungen des Designs an der Wurzel zu packen, muss sich der für das Design Verantwortliche jedoch gegebenenfalls auch mit den einzelnen Elementen der Modellierungssprachen und deren Spezifikation auseinandersetzen. Dabei geht es um die Frage, ob alle notwendigen Elemente für das gewünschte Prozessmodell überhaupt vorhanden sind, ob sie den gewünschten Sachverhalt richtig und unmissverständlich abbilden und ob ihre Einbindung in das Prozessmodell korrekt ist. Danach ist zu entscheiden, ob eine existierende Sprache bzw. Standardsprache verwendet werden kann oder nicht doch eine anwendungsbezogene, die gegebenenfalls erst definiert werden muss. Ist Letzteres der Fall, sind die damit verbundenen Aufgaben klar dem Design zuzuordnen.

3.3 Methodische Grundlagen der Modellierung

Es existiert eine Vielzahl von Modellierungssprachen, die zum Teil sehr unterschiedliche Anforderungen abdecken [AmSo07] [Sinu04] [SoWa07]. Da in den Organisationen die Standardmodellierungssprachen und ihre reine Anwendung im Vordergrund stehen, soll auf den Aspekt der Definition einer (anwendungsbezogenen) Modellierungssprache, die Meta-Modellierung, eingegangen werden. Das Kapitel gliedert sich demnach folgendermaßen:

- *Modellierungssprachen*
- *Standard- vs. anwendungsbezogene Modellierungssprachen*
- *Meta-Modellierung*

3.3.1 Modellierungssprachen

Es wird ein Überblick über allgemein bekannte sowie über eine lehrstuhleigene Modellierungssprachen gegeben.

3.3.1.1 Unified Modeling Language (UML)

Die Unified Modeling Language (UML) [OMG11e] [RaJB04] ist eine universelle, auf die Spezifikation und Dokumentation von Softwareentwicklung ausgerichtete, graphische, objektorientierte Modellierungssprache, die von der Object Management Group (OMG) *standardisiert* wurde. Neben ihrem ursprünglichen Einsatzzweck zur Softwareentwicklung wird die UML auch für die Prozessmodellierung eingesetzt. Von den vierzehn verschiedenen Diagrammtypen, die an und für sich sehr heterogen sind, sich jedoch sprachlich und inhaltlich überlappen [Tesc03], werden dazu insbesondere die Aktivitätsdiagramme (siehe Abbildung 3-5) verwendet.

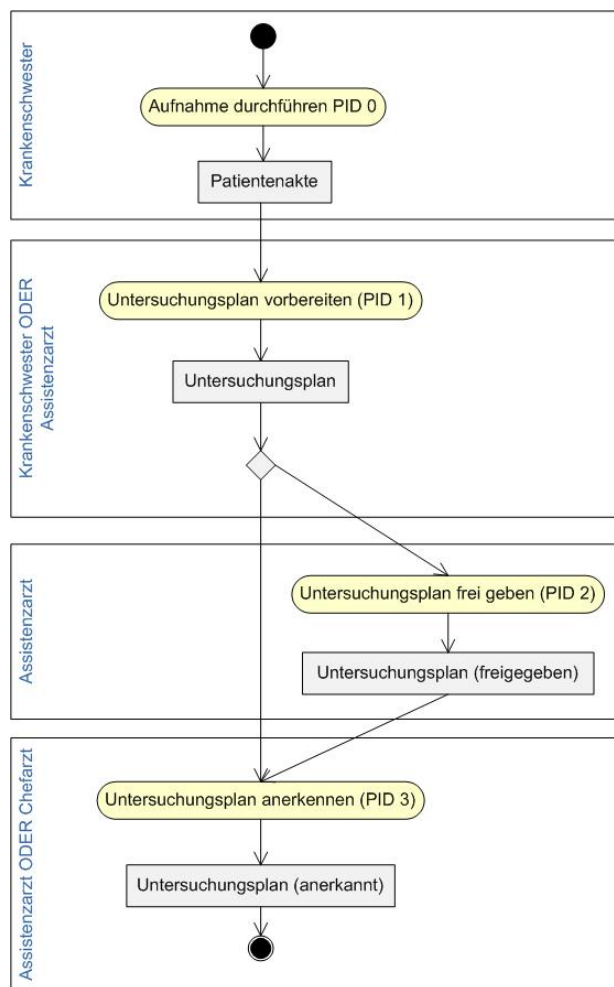


Abbildung 3-5 Beispiel UML-Aktivitätsdiagramm

Mit einem Aktivitätsdiagramm werden Arbeitsabläufe bzw. Workflows mittels elementaren *Aktivitäten* beschrieben. Ein Beispielprozess ist in Abbildung 3-5 zu sehen. Dabei werden folgende Knoten über *Kontroll- und Objektflüsse* sequentiell hintereinander geschaltet:

- **action node (Aktivitäten):** Aktivitäten stellen die auszuführenden Funktionen dar.
- **control node:** Mit dieser Art von Knoten werden Verzweigungen, die Aufspaltung und die Zusammenführung des Kontrollflusses modelliert. Es werden AND und XOR, nicht jedoch OR unterstützt.
- **object node (Daten/ Informationen):** object nodes stellen In- bzw. Output der Aktivitäten dar.

Ein Aktivitätsdiagramm kann andere Aktivitätsdiagramme oder Teile von ihnen referenzieren und unterstützen so die Wiederverwendung durch die Möglichkeit zur hierarchischen Dekomposition.

Kontroll- und Objektfluss werden wie folgt definiert:

- **Kontrollfluss:** Der Kontrollfluss, auch Transition genannt, verbindet die verschiedenen Knotentypen miteinander
- **Objektfluss:** Über die Objektflüsse werden die object nodes (= Daten, Informationen) den action nodes (= Aktivitäten) als In- oder Output zugeordnet

Als ein weiteres Strukturierungselement können Swimlanes verwendet werden. Diesen wird eine organisatorische Einheit zugeordnet und die Aktivitäten, denen die jeweilige Rolle zugewiesen ist.

Obwohl als Standard definiert ist UML auch als anwendungsbezogene Modellierungssprache anzusehen. Basierend auf den wesentlichen Komponenten der Standard-UML (zum Beispiel Aktivitäten bei den Aktivitätsdiagrammen) kann das Vokabular durch die Definition von sogenannten *profile*, als Menge von Stereotypen inklusive Paketimporten, erweitert werden. Diese fügen einem bestimmten Kontext entsprechend den Basiskomponenten neue Eigenschaften zu. Als Beispiel ist *UML 2.0 Superstructure Specification* zu nennen, die vier UML-Profile für Industriestandards im Bereich Softwarekomponenten definiert.

Bewertung

Es handelt sich bei der UML zum einen um einen Standard, der durch seine vielen verschiedenen Diagrammtypen (7 Struktur-, 7 Verhaltensdiagramme) sehr umfassend und auch mächtig ist.

Mit der Möglichkeit zur Definition anwendungsbezogener Aspekte in Form von Stereotypen und Profiles ist dieser Standard zusätzlich flexibel für individuelle Anwendungsfelder anpassbar. Dies ist notwendig, da aufgrund des generischen Sprachansatzes spezielle Konstrukte in den einzelnen Diagrammen nicht vorgesehen sind. Die Konstrukte konzentrieren sich auf den grundlegenden Einsatzzweck der einzelnen Diagramme. Da die ursprüngliche Bedeutung der Konstrukte also nicht verändert werden darf, ist dieser Mechanismus nur eingeschränkt anwendbar. Die Möglichkeiten für designbezogene

Gestaltungsmaßnahmen sind somit gering. Einen Weg, wie dieses Problem behoben werden kann, ist zum Beispiel bei [Volz11] nachzulesen.

Es eignen sich vor allem Aktivitätsdiagramme prinzipiell für die Modellierung von Prozessen, auch wenn sie nicht alle gängigen Aspekte der Prozessmodellierung unterstützen [RvAH06]. So sind der funktionale und datenorientierte Aspekt in Form der activity und object nodes vorhanden, während es an der Definition eines ressourcenbezogenen Aspekts fehlt. „They (die Aktivitätsdiagramme) are not able to capture many of the natural constructs encountered in business process such as cases and notion of interaction with the operational environment in which the process functions“ [RvAH06]. Die Ausdrucksmächtigkeit der Aktivitätsdiagramme im Kontext der Prozessmodellierung ist stark eingeschränkt. Dies kann jedoch durch die Erweiterung um weitere Modellierungskonstrukte in Form von Profiles oder Stereotypen behoben werden. Im Bezug auf solche Erweiterungen sollte jedoch immer der damit stets verbundene, nicht geringe Aufwand bedacht werden.

3.3.1.2 Ereignisgesteuerte Prozessketten (EPK)

Die Ereignisgesteuerte Prozesskette (EPK) [KeNS92] [Köhl06] [Mend08] ist eine semiformale, graphische Modellierungssprache. Ihren Ursprung hat die Sprache in einem gemeinsamen Forschungsprojekt der Universität des Saarlandes und der SAP AG mit dem Ziel der „Dokumentation von Geschäftsprozessen“ [NüRu02] bzw. mit dem Ziel des Designs des SAP R/3 Referenzprozessmodells [CuKe97] [Hern05]. Inzwischen werden sie auch zur Erstellung anderer SAP-unabhängiger Referenzmodelle wie ITIL [IDSS10a] oder SCOR [IDSS10c] verwendet. Die EPKs sind ein wesentlicher Bestandteil der ARIS (Architektur Integrierter Informationssysteme) Plattform zur sichten-orientierten Modellierung von Geschäftsprozessen [Sche00] [Sche02]. Sie werden dabei vor allem in kleineren und mittleren Unternehmen eingesetzt.

Eine offizielle Standardisierung ist bisher nicht erfolgt. Es handelt sich vielmehr um einen „de-facto“ Standard. Dies ist auch ein Grund dafür, dass die Verbreitung außerhalb des deutschsprachigen Raums etwas geringer ausfällt. Nichtsdestotrotz gibt es eine Vielzahl internationaler Forschungsprojekte, die EPKs verwenden bzw. weiter entwickeln (siehe dazu zum Beispiel [LaRo08]).

EPKs stellen Prozesse als gerichteten Graph bestehend aus *Ereignissen*, *Funktionen*, *Konnektoren* dar, wie es beispielhaft in Abbildung 3-6 zu sehen ist.

- **Ereignisse:** EPKs starten und enden mit mindestens einem Ereignis. Es handelt sich um Zustände, die Funktionen auslösen (triggern) bzw. deren Vollendung signalisieren.
- **Funktionen:** Funktionen stellen Aufgaben bzw. Prozesse dar, die ausgeführt werden sollen; sie folgen immer auf ein Ereignis.
- **Konnektoren:** Konnektoren dienen dem Aufspalten und Zusammenführen von alternativen (XOR, OR) bzw. parallelen (AND) Kontrollflüssen.

Eine Anreicherung der ursprünglichen Sprache stellen die erweiterten EPKs (eEPKs) dar [Sche99]. Dem Meta-Modell der EPKs wurden dazu weitere spezielle Symbole hinzugefügt. Damit kann die prinzipiell stark funktional ausgerichteten EPK unter anderem auch *Rollen*, *technischen Ressourcen*, *Daten* (= Informationen), *Material* und *Leistungen* (zum Beispiel Produkte) abbilden. Daten sowie Rollen sind auch in Abbildung 3-6 zu sehen.

- **Organisatorische Einheiten:** Die organisatorischen Einheiten (Rollen, Gruppen, Mitarbeiter) dienen der Darstellung der Gliederungsstruktur eines Unternehmens. Sie werden direkt einer entsprechenden Funktion zugeordnet (siehe dazu auch [Buss98]).
- **Informations- und Ressourcenobjekte:** Mit dieser Art von Objekten erfolgt die Abbildung eines Gegenstandes aus der realen Welt. Die Objekte werden als In- oder Output einer Funktion definiert.

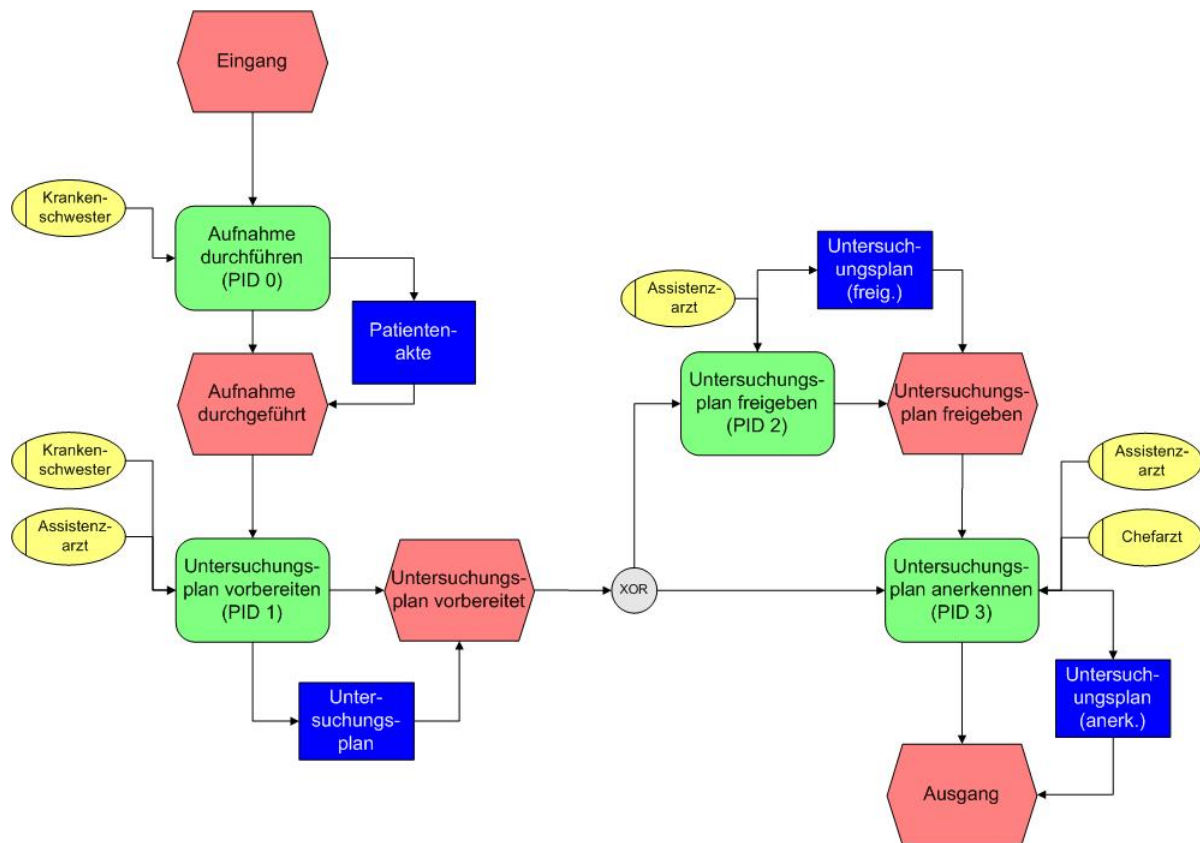


Abbildung 3-6 Beispiel EPK-Modell

Bewertung

Die EPKs sind gezielt zur Modellierung von Geschäftsprozessen entwickelt worden, so dass dabei keine Zweckentfremdung erfolgt, wie es bei anderen Modellierungssprachen, zum Beispiel bei den Aktivitätsdiagrammen der UML, zu sehen wäre. Durch die geringe Anzahl

an Konstrukten und auch aufgrund der einfachen Syntax ist diese Modellierungssprache sehr leicht zu erlernen [ADEF04].

Der generische Ansatz dieser Modellierungssprache zeigt sich in den sehr allgemeingültig definierten Konstrukten. Da weder Syntax noch Semantik der EPKs präzise festgelegt sind, besteht bei den damit erstellten Prozessmodellen ein großer Interpretationsspielraum [KeNS92] [vAal99], der nicht unbedingt positiv zu bewerten ist. Gerade bei der maschinellen Verarbeitung solcher Prozessmodelle, aber auch in der Entwurfsphase von Softwareentwicklungsvorhaben kann diese Freiheit aufgrund von fehlender oder nicht möglicher Interpretationen zu Problemen führen [Ritt00].

Der Schwerpunkt der EPKs liegt auf dem betriebswirtschaftlichen Kontext. Vor allem standardisierte Abläufe mit ihren formalen und geordneten Strukturen lassen sich damit gut beschreiben. In anderen Anwendungsbereichen, wie zum Beispiel dem medizinischen Kontext, oder aber zur Abbildung kreativer und komplexer Prozesse, ist die Anwendung weniger zu empfehlen.

Im Gegensatz zur UML gibt es bei den EPKs keine Möglichkeit, anwendungsbezogene Inhalte bzw. Aspekte zu definieren. Die Erweiterung der EPKs hin zu eEPKs beschränkt sich auf den betriebswirtschaftlichen Standardkontext. Lediglich Ressourcen- und Informationsobjekte sowie organisatorische Einheiten wurden ergänzt. Maßnahmen im Kontext des Designs können sich also lediglich auf die Formulierung von Modellierungsregeln beziehen, nicht jedoch auf die Sprachdefinition selber.

3.3.1.3 Business Process Modeling Notation (BPMN)

Die Business Process Modeling Notation [OMG11a] ist ein relativ junger Standard zur graphischen Modellierung von Prozessen. Ursprünglich im Jahre 2000 von Stephen A. White entwickelt und seit 2005 in Verantwortung der OMG (Object Management Group) erfolgt die Verabschiedung als Standard im Jahre 2006. Die neueste Version BPMN 2.0 wurde 2011 verabschiedet.

Die grundlegende Motivation für die Entwicklung von BPMN war bzw. ist folgende: „A standard graphical notation will facilitate the understanding of the performance collaborations and business transactions within and between the organizations“ [OMG06]. Der Standard richtet sich sowohl an Prozessanalysten, Anwender von Geschäftsprozessen als auch an technische Entwickler zum Zweck der Dokumentation und Kommunikation von Geschäftsprozessen und darauf aufbauend auch zur Entscheidungsunterstützung [FrRH10] [tHAA10]. BPMN-Modelle werden dabei auch zur Spezifikation der Anforderungen von Softwaresystemen eingesetzt.

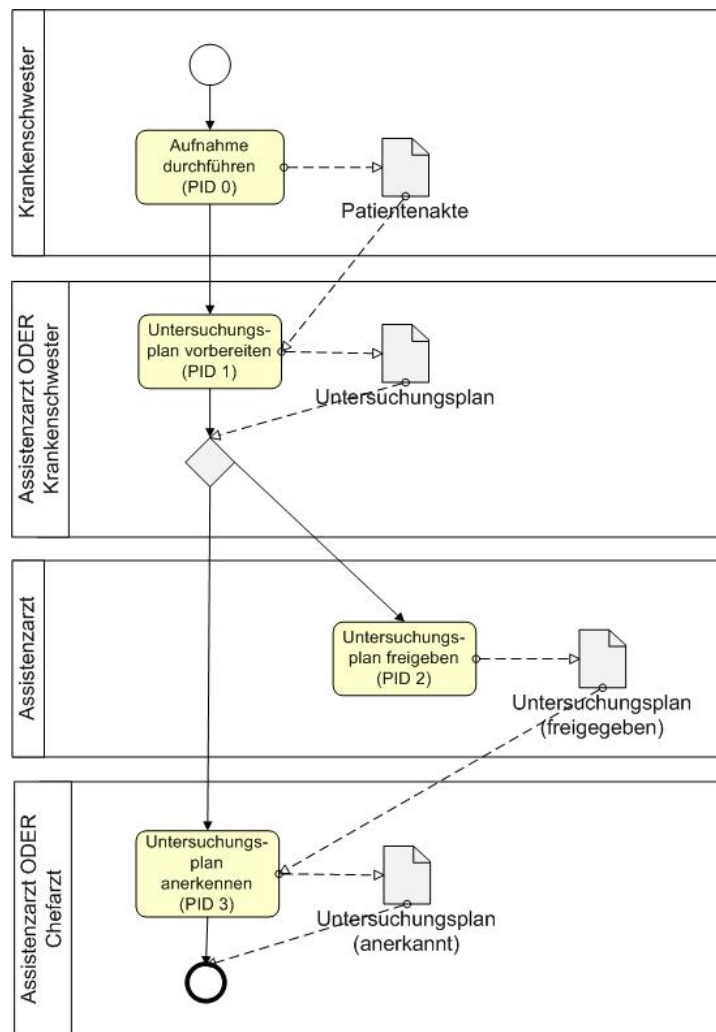


Abbildung 3-7 Beispiel BPMN-Modell

Die Hauptmodellierungselemente (flow objects) von BPMN sind folgende:

- **Activities:** Es werden zwei Arten von *activities* (= Aktivitäten) unterschieden. Elementare *activities* stellen tasks (=Aufgaben) dar, die nicht weiter unterteilt werden. Aktivitäten können aber auch zu komplexen activities zusammengefasst werden und werden dann als „subprocess“ bezeichnet.
- **Gateway:** Bei den *gateways* handelt es sich um Modellierungselemente zur Verzweigung und zum Zusammenführen des Kontrollflusses. Es werden XOR-, OR- und AND-Gateways unterschieden.
- **Event:** *Events* stellen Ereignisse im Ablauf des modellierten Geschäftsprozesses dar. Sie lassen sich in drei Klassen unterscheiden: Start-, Zwischen- und Endereignis. Das *start event* stellt den Beginn eines Prozesses dar, das *end event* den Abschluss. *Intermediate events* erfassen alle Ereignisse, die zwischen Start und Ende eines Prozesses liegen.

Zur vollständigen Abbildung dieser flow objects existieren spezielle Verbindungsobjekte. Der *sequence flow* legt die Reihenfolge der activities fest. Der *message flow* zeigt den Austausch von Nachrichten (*messages*) zwischen den Aktivitäten bzw. den dazugehörigen Geschäftsprozessteilnehmern. Da die Nachrichten der Darstellung der Kommunikation zwischen den Teilnehmern dienen, gibt es seit der Version BPMN 2.0 nun auch Daten (*data objects*) [FrRH10]. Diese repräsentieren Informationen, die durch den Prozess fließen. Des Weiteren gibt es Assoziationen, über die eine Verbindung von Artefakten (zusätzliche Informationen zu dem Gesamtprozess) und einem flow object definiert werden kann.

Der organisatorische bzw. operationale Aspekt wird durch pools und lanes dargestellt, welche vom Prinzip her vergleichbar mit den Swimlanes der EPKs sind. *Pools* stellen eine Art graphischen Container für eine Menge von activities dar, die alle einem *participant* zugeordnet sind, zum Beispiel einer Abteilung oder aber auch einem System. Zur weiteren Kategorisierung und Unterteilung der Aktivitäten in einem Pool, zum Beispiel bei den Abteilungen zusätzlich nach Rollen, sind die *lanes* vorgesehen.

Eine direkte informationstechnisch unterstützte Ausführung der Modelle ist derzeit nicht vorgesehen bzw. möglich. Um dennoch eine Brücke zwischen dem Design bzw. der Modellierung und der Implementierung zu schlagen, ist zum Beispiel die Transformation in die ausführbare Business Process Execution Language for Web Services (BPEL4WS) [IBM11] möglich [Whit04] [OvAD06] oder auch in die XML Process Definition Language (XPDL) [WoMC11].

Bewertung

BPMN ist prinzipiell als eine alternative Modellierungssprache zu EPKs oder den UML Aktivitätsdiagrammen zu sehen. Da BPMN erst später als die beiden anderen Sprachen spezifiziert wurde und somit davon nicht unbeeinflusst bleiben konnte, wurden bei BPMN einige Verbesserungen eingearbeitet [tHAA10] [RRIG06].

Doch auch wenn BPMN speziell für die Modellierung von Prozessen entwickelt wurde, können einige Aspekte bzw. Strukturen, die mit dem Thema Prozess eigentlich in einer sehr engen Verbindung zu sehen sind, nicht abgebildet werden [FrRH10]: Prozesslandschaften, Aufbauorganisation, Strategien, Geschäftsregeln und IT-Landschaften. Diesem Kritikpunkt wird jedoch mit dem Argument entgegen getreten, dass sich BPMN auf Prozesse in ihrem eigentlichen Sinn, nämlich als eine reine zeit-logische Abfolge von Aktivitäten konzentriert und daher solche erweiterten Betrachtungsweisen in den Hintergrund rücken.

BPMN als eine Standardsprache bietet keine Möglichkeit, anwendungsbezogene Inhalte bzw. Aspekte zu definieren. Es können zwar Artefakte nicht mehr nur als freie Anmerkung spezifiziert werden, sondern auch als eigene Symbole, doch können diese lediglich über Assoziationen mit Flussobjekten verbunden werden und dienen mehr als Hinweis [FrRH10].

Sie dürfen nicht in den Kontrollfluss integriert werden. Die in Kapitel 3.1.1 eingeführten Aufgaben des Designs können also nicht vollständig realisiert werden.

Die Studien von [RIRG06] und [Reck10] zeigen Schwachstellen der Modellierungssprache auf. Unter anderem können mit BPMN keine Zustände, sondern nur Aktivitäten und Ereignisse dargestellt werden. Sachverhalte können mit verschiedenen Modellierungskonstrukten dargestellt werden, da diese semantisch nicht eindeutig definiert sind. So können Rollen mit „Pools“ und „Lanes“ modelliert werden. Die beiden Konstrukte „Pools“ und „Lanes“ sind mit mehrfachen Bedeutungen überladen und können somit nicht eindeutig verwendet werden. Es gibt eine sehr differenzierte Menge an „Events“ mit jeweils spezifischer Bedeutung. Angesichts der Menge von Events ist jedoch Erklärungs- bzw. erhöhter Einarbeitungsbedarf vorhanden [Allw10].

BPMN dient vor allem der Modellierung von Geschäftsprozessen auf konzeptioneller Ebene. Für Modelle, die direkt mittels einer Workflow Engine ausgeführt werden sollen, bietet sich BPMN weniger gut an, was jedoch auch nicht das primäre Ziel war bzw. ist [tHAA10] [Whit04]. Dies zeigt sich auch in dem dafür zusätzlich notwendigen Modelltransformationsschritt, der einen zusätzlichen Aufwand erfordert.

Trotz der genannten Kritikpunkte wird der Standard BPMN immer häufiger bei den Modellierungsprojekten der Organisationen verwendet [Allw10]. Die Gründe dafür liegen zum einen in der Standardisierung, die ein verlässliches Rahmenwerk vorgibt, aber wohl auch an der fehlenden Konkurrenz.

3.3.1.4 Aspektororientierte Prozessmodellierung

Die aspektororientierte Modellierung (AOPM, oder engl. POPM = Perspective Oriented Process Modeling) wurde im Rahmen des MOBILE Projekts [Mobi00] an der Friedrich-Alexander-Universität Erlangen-Nürnberg entwickelt. Dabei standen zunächst primär die Modellierung von Workflows im Vordergrund und weniger die im Rahmen dieser Arbeit fokussierten Geschäftsprozesse. Das grundlegende Konzept für dieses Projekt wurde erstmals in [Jab194] vorgestellt, sowie in [Jab195] und [JaBu96] ausführlich beschrieben.

Es gibt sowohl eine textuelle, als auch eine graphische Spezifikation dieser Modellierungssprache [Jab101], wobei Letztere im Folgenden im Fokus stehen soll. Basierend auf den Projekterfahrungen (u.a. [Horn03]) wurde von der Firma Prodato das Prozessmodellierungswerkzeug i>pm (= Integrated Process Modeler) [Prod05] entwickelt. Dieses adaptiert das Konzept der aspektororientierten Workflow-Modellierung für Anwendungsprozesse und kann somit auch für die Modellierung von Geschäftsprozessen verwendet werden. Inzwischen gibt es mit OMME (Open Meta Modeling Environment, [VoJa10a] [VoJa10b]) einen ersten Prototypen einer grundlegend überarbeiteten Version des Modellierungswerkzeuges (mehr Informationen folgen im weiteren Verlauf der Arbeit).

Die Grundidee vom AOPM ist, dass ein Modellierungskonstrukt aus verschiedenen, voneinander unabhängigen, orthogonalen Teilen besteht. Diese verschiedenen Aspekte ermöglichen eine umfassende Beschreibung des Modellierungsgegenstandes. Bei einem konkreten Anwendungsfall müssen jedoch nicht zwingend alle definierten Aspekte verwendet werden. Konzeptionell ist die Anzahl der Aspekte beliebig; es können jederzeit neue Aspekte hinzugefügt werden. Ebenso können die Aspekte an sich jederzeit erweitert werden. Die Basis für diese Erweiterungen bildet die der Methode zugrunde gelegte Meta-Modell-Hierarchie nach [AkKü05], [HeGo06] und [JaVD09].

Eine weitere wesentliche Eigenschaft dieser Modellierungssprache ist das sog. Typ-Verwendungskonzept. Dies bedeutet, dass ein Modellierungskonstrukt einmalig definiert wird (= Typ) und dann beliebig oft in dem zu erstellenden Prozessmodell wiederverwendet werden kann (= Verwendung). So wird zum Beispiel zunächst der Daten-Typ „Untersuchungsplan“ einmal definiert und dann im Beispielprozess in Abbildung 3-8 dreimal verwendet.

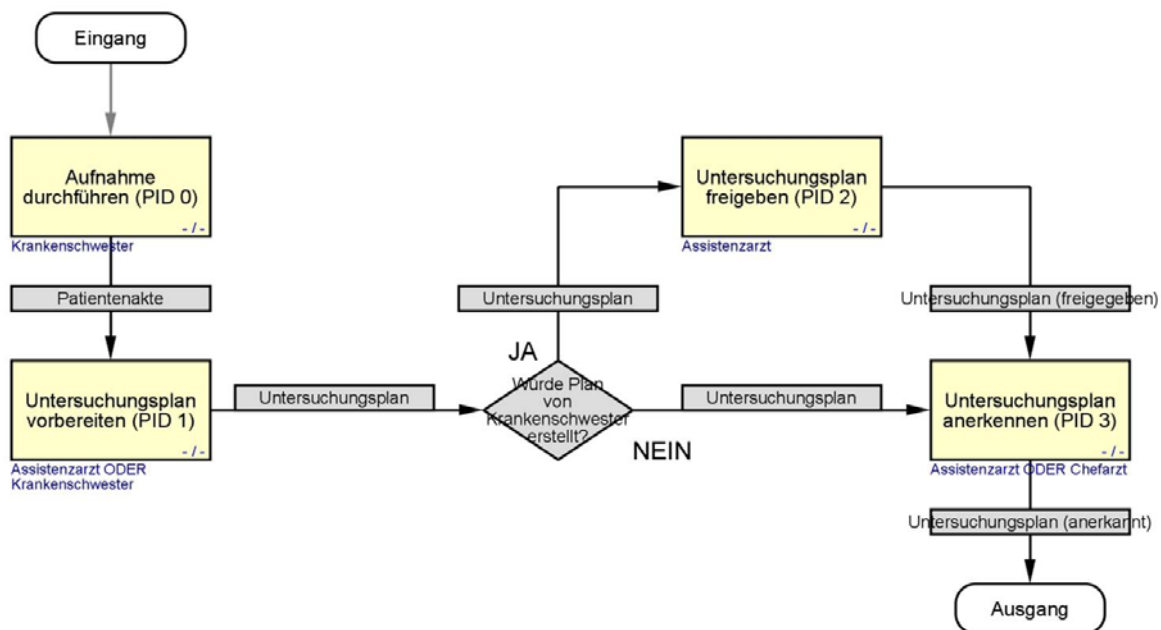


Abbildung 3-8 Beispiel AOPM-Modell

Nach [JaBu96] werden zumeist die folgenden Aspekte bei der Prozessmodellierung verwendet:

- **Funktionaler Aspekt:** Zur Festlegung einer grundlegenden Struktur werden Prozesse und Unterprozesse definiert. Kann ein Prozess nicht mehr in weitere Teilprozesse aufgegliedert werden, handelt es sich um einen elementaren Prozess; andernfalls spricht man von einem kompositen Prozess.

- **Verhaltensbezogener Aspekt:** Die einzelnen Prozesse werden mittels Kontrollfluss in eine Reihenfolge gebracht. Der Kontrollfluss kann linear oder verzweigt sein, parallel oder alternativ erfolgen.
- **Organisatorischer Aspekt:** Es wird festgelegt, wer für die Ausführung eines Schrittes verantwortlich ist, wobei lediglich neutrale Rollen definiert werden, statt konkrete Personen anzugeben (siehe Details dazu in [Buss98], [JaTa09] und [TaVJ10]).
- **Operationaler Aspekt:** Es wird angegeben, mit Hilfe welcher Werkzeuge und Systeme die eigentliche Funktionalität erbracht wird, ob zum Beispiel mit Microsoft Word und Excel oder speziellen medizinischen Geräten.
- **Datenorientierter Aspekt:** Dieser Aspekt beschreibt den Datenfluss von einem Prozessschritt zum nächsten. Ein Datenfluss entsteht dabei durch eine Output-Input-Beziehung zwischen zwei Prozessen. Während der eine Prozessschritt das Datum produziert, konsumiert der nachfolgende Prozessschritt dieses Datum.

Als weitere Aspekte sind der historische [Schl04], der qualitäts- sowie der sicherheitsbezogene Aspekt [JaBu96] zu nennen.

Bewertung

Die AOPM definiert grundlegende Konstrukte zur Modellierung von Geschäfts- wie auch Workflowprozessen. Durch die begrenzte Anzahl an leicht interpretierbaren Modellierungskonstrukten in der Grundversion von AOPM (domänenspezifische Konstrukte sind dabei nicht mit eingeschlossen) ist die Sprache für den Anwender leicht zu erlernen.

Alle wichtigen Konstrukte und Eigenschaften sind spezifiziert, wodurch die Sprache sehr mächtig ist. Durch die Definition der verschiedenen Aspekte und die Zuordnung der Konstrukte zu diesen, ist eine klare Strukturierung der Modellierungssprache wie auch der damit erstellten Modelle möglich. Aufgrund der aspektorientierten Komposition und der damit verbundenen Modularisierung ist zudem ein hohes Maß an Flexibilität der Modellierungssprache vorhanden. Es kann eine beliebige Anzahl an Aspekten definiert werden, die auf der einen Seite unabhängig voneinander sind, sich auf der anderen Seite jedoch gegenseitig ergänzen. Durch die jederzeit mögliche Erweiterung um zusätzliche Konstrukte, Attribute und/ oder Aspekte kann die AOPM leicht individuell an spezielle Domänen angepasst werden. So existieren inzwischen mit den Versionen i>PM4Med bzw. i>PM4QM Anpassungen für die medizinische Domäne bzw. für die Aspekte des Qualitätsmanagements. Die in Kapitel 3.1.1 für das Design erläuterten Handlungsmöglichkeiten können bei AOPM vollständig umgesetzt werden. Dies ist ein entscheidender Vorteil gegenüber den in den vorherigen Kapiteln vorgestellten Modellierungssprachen.

Das Prinzip der Wiederverwendung in Form des Typ-Verwendungskonzepts ist als eine herausragende Eigenschaft dieser Modellierungssprache zu bewerten. Es trägt beim Modellierungsvorgang wesentlich zur Effizienzsteigerung bei, aber auch zur Reduktion von Fehlern bei der Modellierung.

3.3.2 Standard- vs. Anwendungsbezogene Modellierungssprachen

Wie in den Ausführung in Kapitel 3.3.1 zu sehen war, kann grundlegend zwischen Standardsprachen, wie BPMN, und anwendungsbezogenen Sprachen, zum Beispiel AOPM, unterschieden werden. Der wesentliche Unterschied besteht darin, dass Letztere individuell für einen konkreten Anwendungsbereich- bzw. fall erstellt sind. Wie diffizil allein diese Entscheidung bereits ohne konkret vorliegenden Anwendungsfall ist, zeigt folgender Abschnitt.

3.3.2.1 Standardmodellierungssprache

Vorteile

Für die Standardmodellierungssprachen spricht ihre *Allgemeingültigkeit bzw. Generizität*. Da diese nicht die Charakteristika einer speziellen Anwendungsdomäne berücksichtigen, sind sie so allgemein definiert, dass sie die bei den meisten Anwendungsfällen vorzufindenden Eigenschaften und Aspekte generisch abdecken [FiHR08] [Meil05] [Wats08] und einen großen Freiheitsgrad bei der Modellierung bieten [Allw06]. So kann zum Beispiel die Standardmodellierungssprache BPMN für die Modellierung von Geschäftsprozessen eines Industrieunternehmens verwendet werden, aber ebenso zur Modellierung klinischer Pfade (= medizinischer Prozesse). Anwendungsbezogene Modellierungssprachen können zwar auch derart verwendet werden. Dies kann jedoch dazu führen, dass Modellierungselemente nicht in ihrem eigentlichen Sinne verwendet werden, so dass die Eindeutigkeit der Sprache leidet.

In einem Abstimmungsprozess zwischen den Verantwortlichen und möglichen Anwendern der Standardmodellierungssprache werden allgemeingültige Sprachelemente und eine allgemeingültige Terminologie definiert [DeIN10] [GrKo05]. Damit sind Standardsprachen für den normalen Anwender üblicherweise leichter zu verstehen (*allgemeine Verständlichkeit*), weil kein spezielles domänenspezifisches Wissen zum Verständnis nötig ist. Die Modelle können somit als Basis zur Kommunikation zwischen den verschiedenen Stakeholdern (dies sind Manager des Unternehmens, Interessenten anderer Organisationen oder auch Kunden) verwendet werden [SmFi03], die keine ausgewiesenen Kenntnisse im Bereich der Modellierung besitzen oder wenn es bei der Kommunikation um domänenübergeordnete Aspekte, wie zum Beispiel strategische Entscheidung innerhalb einer Organisation geht. Der Aspekt der besseren Verständlichkeit gilt jedoch nicht uneingeschränkt. Zum Zweck der Standardisierung muss teils auch in solchem Ausmaß abstrahiert werden, dass der Anwender keinen direkten Zugang mehr zu der Sprache bzw. dem damit dargestellten Sachverhalt findet [ClSW08] [Jabl09] [Wats08].

Die Standardmodellierungssprachen sind oftmals kompatibel und können von verschiedenen (Modellierungs-) Systemen interpretiert werden (*Interoperabilität bzw. Probabilität*), so dass sie vielseitig einsetzbar sind [DeIN10] [GrRe93] [Jabl09] [VoZe09] [Wats08]. Die Wiederverwendbarkeit erspart Aufwand bzgl. Schnittstellenbeschreibungen und Konvertierungstools sowie bzgl. der Entwicklungen von Individuallösungen. Es hat zudem auch eine positive Auswirkung auf den Bekanntheitsgrad einer Modellierungssprache. Die Standardmodellierungssprache BPMN wird zum Beispiel von den Modellierungswerkzeugen ARIS Toolset, dem Interstage Business Process Manager 7.1 (Fujitsu), Signavio Process Editor (Signavio GmbH) sowie SAP NetWeaver Composition Environment (SAP) implementiert; aktuell gibt es 71 Lösungen [OMG11b].

Da Standards prinzipiell einen höheren Bekanntheitsgrad haben als domänenspezifische Sprachen, fallen die *Kosten* der Einarbeitung geringer aus oder entfallen sogar komplett [Wats08]. Dieser Zusammenhang ist damit zu erklären, dass durch den Bekanntheitsgrad viele potentielle Anwender bereits Kenntnisse über die Sprache haben, so dass der zusätzliche Schulungsaufwand und folglich die Kosten geringer ausfallen. Aus Sicht des Anwenders entfallen zudem die Kosten für die Entwicklung solcher Sprachen. Diese tragen die für die Entwicklung und Pflege der Standards zuständigen Organisationen, wie zum Beispiel inzwischen die Object Management Group OMG für BPMN. Offene Standards werden auch ohne Lizenzgebühr angeboten [SmFi03].

Die *Verantwortung für die Entwicklung und Pflege der Standards ist klar definiert* und liegt in den Händen einer einzigen verantwortlichen Organisation. Damit können Unstimmigkeiten durch verschiedene Entwicklungsgruppen vermieden werden. So wird BPMN wie bereits erwähnt seit 2005 von der Object Management Group gepflegt. Zudem nimmt diese Organisation dann auch den hohen Kosten- und Zeitaufwand der Entwicklung und Pflege auf sich, wovon gerade kleinere Organisation mit geringen finanziellen Mitteln profitieren.

Standardsprachen bieten sich als Basis zur Entwicklung domänenspezifischer Sprachen an [GiGW00]. Sie bilden die Grundlage für die Ergänzung um spezielle Aspekte. Damit ergänzen sich Standard- und domänenspezifische Sprachen und es zeigt sich hier sehr deutlich, dass beide Ansätze ihren Sinn und ihre Berechtigung haben.

Nachteile

Bemängelt wird unter anderem, dass Standardmodellierungssprachen inhaltlich oftmals *nicht vollständig* sind [GrKo05] [Jabl09] [RIRG06]. Aufgrund ihrer Allgemeingültigkeit und Generizität können die Sprachen nicht alle Facetten der möglichen Anwendungsfälle abdecken [GrKo05] [Meil05] [Müll90]. So konnten zum Beispiel konkret für BPMN empirische Untersuchungen das Fehlen von wesentlichen Aspekten aufdecken. Dort fehlt es unter anderem an Möglichkeiten zur Darstellung der Struktur eines Systems [RIRG06] sowie von Geschäftsregeln (= business rules) [Reck10].

Im Bewusstsein des Problems der mangelnden Vollständigkeit, wird versucht, möglichst viele Aspekte in einen Standard zu integrieren. Dies führt jedoch unweigerlich dazu, dass sich die Anzahl der Konstrukte und Funktionalitäten und damit der Umfang der Standards erhöht und diese *inhaltlich wie funktional überladen* sind. Die Produktivität der Modellierung leidet dabei und die Sprachen werden unhandlich. So ist für BPMN eine Vielzahl verschiedener sog. „Events“ definiert (siehe Anhang B). Wie auch in [RIRG06] und [Reck10] bestätigt werden konnte, ist es leicht nachvollziehbar, dass ein Anwender von den zur Verfügung stehenden Möglichkeiten erschlagen ist, auch wenn jedes der Konstrukte für sich verständlich ist. Durch die damit angestrebte Vollständigkeit geht im Bezug auf die einzelnen Elemente jedoch die mit Standardmodellierungssprachen angestrebte Generizität verloren.

Da die Spezifikation der Standardmodellierungssprache allgemeingültig gehalten ist [Meil05] [Müll90], leidet oftmals die Präzision der Sprache. Dies führt zum *Verlust der Mächtigkeit* bzw. wie [CISW08] es formuliert: “any gain made in widening the applicability of a language to different domains will be at the expense of the richness of the language that makes it so suitable for a particular domain”. Eine kompakte, fachspezifische Formulierung eines Sachverhaltes ist nicht möglich [DSMF10] [FiHR08] [FrvL03]. Die domänenspezifischen Anforderungen an eine Sprache sind jedoch zum Teil so gegensätzlich, dass sie nur schwer allgemeingültig „unter einen Hut“ gebracht werden können [CISW08] [Jabl09]. Die Konstrukte *Lane* und *Pool* sind in BPMN sehr undifferenziert definiert und somit ihr konkreter Einsatzzweck für den Anwender nur schwer ersichtlich oder sogar missverständlich [Reck10].

Eng verbunden mit der allgemeingültig gehaltenen Definition sind Standardmodellierungssprachen nicht an die Erfordernisse einer speziellen Anwendungsdomäne angepasst und daher dann bei anwendungsbezogenen Fällen nur *eingeschränkt anwendbar* [FrvL03] [Meil05] [Müll90]. Es fehlen die Konstrukte oder dazugehörige Attribute zur Darstellungen gewisser Aspekte eines speziellen, aber in dem vorliegenden Fall essentiellen Sachverhaltes. Beispielhaft zu nennen wäre hier die Angabe einer Kosten-Kennzahl an einem Prozessschritt.

Die Standardisierung stellt einen sehr langwierigen, zeit- und kostenintensiven Prozess dar, wie auch die sich anschließende Pflege, um den Standard immer aktuell zu halten bzw. mögliche Verbesserungen einzubringen [CISW08]. Wenn es sich nicht um einen offenen Standard handelt, fallen für den Anwender Lizenzkosten an, die im kommerziellen Bereich meist sehr hoch sind [GrKo05].

3.3.2.2 Anwendungsbezogene Modellierungssprachen

Vorteile

Im Bezug auf das genau abzugrenzende Anwendungsgebiet bzw. den Einsatzzweck haben die anwendungsbezogenen Modellierungssprachen durch die speziellen Konstrukte und deren Funktionalitäten eine hohe *Ausdrucksmächtigkeit* [Allw06] [FiHR08] [MeHS05] [vDKV00].

Der abzubildende Sachverhalt kann sehr präzise sowie eindeutig modelliert werden [FiHR08] [Wats08]. Dies wirkt sich positiv auf die Produktivität des Modellierungsvorgangs aus [MeHS05] [ToKe04] [vDKV00], da Diskussionen darüber, wie ein Sachverhalt eindeutig dargestellt oder in der Modelldarstellung verstanden werden kann, erheblich reduziert werden können. Da mit der Verwendung einer anwendungsbezogenen Modellierungssprache eine genauere Abbildung des Anwendungsbereichs gemäß den gestellten Anforderungen möglich ist [FiHR08] [LPZE07], erhöht sich damit prinzipiell (jedoch nicht automatisch) die Qualität der damit erstellten Modelle [ToKe04]. So wurde für AOPM der sog. „Evidenzbasierte Entscheider“ entwickelt [Meil05]. Er dient der Repräsentation evidenzbasierter medizinischer Entscheidungen in Klinischen Pfaden. Diese müssen nicht als Einzelentscheidungen modelliert werden, sondern können über das domänenspezifische Konstrukt zusammengefasst werden. Dabei kann das Konstrukt eine Reihe von Kriterien abbilden, auf denen die Entscheidung beruht.

Die anwendungsbezogenen Modellierungssprachen sind mit Konstrukten und speziellen Eigenschaften und Funktionen *nicht überladen* [ToKe04] [LPZE07]. Es wird nur das spezifiziert, was für die abgegrenzte Anwendungsdomäne relevant ist. Eine Kapselung der Komplexität wird durch den der Domäne entsprechenden Abstraktionsgrad möglich [ToKe04] [DSMF10]. Denn jede Domäne hat ihre eigenen, individuellen Konzepte und Abstraktionsprinzipien [ToKe04]. Ein Beispiel wäre auch hier der Evidenzbasierte Entscheider [Meil05], der eine Vielzahl von medizinischen Einzelentscheidungen und deren Entscheidungskriterien in einem Konstrukt zusammenfasst.

Domänenspezifische Sprachen stellen oftmals eine *Weiterentwicklung von Standards* dar [GiGW00] [ToKe04] [Wats08]. Damit reduziert sich die zeit- und kostenintensive Grundlagenentwicklung. Es kann direkt mit der Konzeption und Ausarbeitung der erweiterten anwendungsbezogenen Aspekte begonnen werden, wobei auch dies im Bezug auf den Aufwand nicht unterschätzt werden darf [Wats08]. Ein Beispiel wäre die anwendungsbezogene Erweiterung der UML über sog. profiles, die aus Stereotypen und Importpaketen definiert werden [OMG11e]. Ein profile ist eine Erweiterung des UML2-Metamodells und wird für spezielle Anwendungsdomänen erstellt. Auch die Ereignisgesteuerten Prozessketten (EPK) können hier als Beispiel genannt werden, auch wenn es sich nicht um einen offiziell verabschiedeten Standard handelt. Allgemein anerkannt wird diese Modellierungssprache im wissenschaftlichen Bereich zum Beispiel erweitert für den Aspekt der Konfiguration zur c-EPC (configurable Event driven Process Chain). Näheres dazu ist in [LaRo08] oder [LLSD07] zu finden. In der Version 2.0 ist nun auch der Standard BPMN individuell erweiterbar. Bis zu einem gewissen Grad geht damit der Vorteil einer Standardmodellierungssprache verloren, was jedoch durch die Vorteile der individuellen Anpassung ausgeglichen werden kann.

Die anwendungsbezogenen Modellierungssprachen sind dadurch gekennzeichnet, dass ihre Konzepte und Symbole direkt mit Begriffen bzw. der Semantik korrespondieren, die dem Anwender vertraut sind [FrvL03] [ToKe04]. Die Modelle sind für den Anwender somit

leicht(er) verständlich, da sie in seiner „(fachlichen) Muttersprache“ verfasst sind [ToKe04] [vDKV00].

Nachteile

Die Entwicklung einer anwendungsbezogenen Sprache ist mit einem *hohen Zeit- und Kostenaufwand* für die Verantwortlichen verbunden, auch wenn als Grundlage eine bereits ausgearbeitete Standardsprache verwendet werden kann (die Entwicklung einer solchen ist ebenfalls mit einem hohen Entwicklungsaufwand verbunden, wie es bei den Contra-Argumenten der Standardsprachen zu lesen war). Zur Entwicklung einer domänenspezifischen Modellierungssprache sind sowohl Kenntnisse bzgl. der Entwicklung einer Sprache sowie spezielles domänenspezifisches Wissen notwendig [LPZE07] [MeHS05]. Da für die Anwendung der anwendungsbezogenen Sprache auch das entsprechende Fachwissen notwendig ist, müssen die Anwender explizit in diese Modellierungsmethoden eingeführt werden, was mit zusätzlichem Aufwand hinsichtlich Zeit und Kosten (nicht direkt im Rahmen der Entwicklung, aber im Anschluss daran) verbunden ist.

Die *Anwendbarkeit* einer anwendungsbezogenen Modellierungssprache ist auf die entsprechende Domäne *begrenzt* [vDKV00]. Es können die typischen Aufgaben aus der entsprechenden Domäne gelöst werden [LPZE07]. Die Lösungen können jedoch nicht auf Probleme in anderen Domänen übertragen werden. So ist *i<pm4med* speziell auf den medizinischen Kontext abgestimmt. Wie bereits mehrfach erwähnt, wurde der sog. „Evidenzbasierte Entscheider“ [Meil05] als zusätzliches Modellierungskonstrukt konzipiert. Die Erfahrung hat gezeigt, dass sich dieses Konstrukt zur Modellierung komplexer Entscheidungen in anderen Domänen nur ungenügend verwenden lässt. Ein Ansatz zur Problemlösung wäre, das Element bei der Verwendung außerhalb der dazugehörigen Domäne „wegzulassen“. Dann sollte das spezielle Konstrukt aber komplett für den Anwender verborgen sein und nicht nur von der Verwendung ausgeschlossen werden, um den Anwender mit möglichst wenig Konstrukten zu konfrontieren. Oftmals ist dieses „Weglassen“ jedoch mit zusätzlichen Implementierungsschritten an dem Modellierungswerkzeug verbunden, die Zeit kosten und zusätzliche Ressourcen beanspruchen.

Für *Nicht-Domänenexperten* sind die domänenspezifischen Sprachen und die dazugehörigen Modelle *schwerverständlich*, da ihnen das dazu nötige Wissen fehlt. Sie können zum einen den anwendungsbezogenen Modellinhalt an sich nur schwer verstehen, zum anderen auch die Darstellungsform, die sich am Abstraktionsverständnis der Domäne orientiert. Dies kann gerade bei (strategischen) Entscheidungen, an denen die Nicht-Domänenexperten beteiligt sind, negative Konsequenzen haben, da sich die entscheidungsrelevanten Informationen möglicherweise nicht vermittelt lassen. Es müssen also entsprechende Erklärungen gegeben werden oder eine für die Nicht-Domänenexperten verständliche Darstellungsform zusätzlich erstellt werden. Beides ist mit zusätzlichem Aufwand verbunden und kann zudem schnell zu Missverständnissen führen. Dies ist eindeutig als ein Nachteil der domänenspezifischen Sprachen anzusehen.

3.3.3 Meta-Modellierung

Metamodelle dienen, stark vereinfacht, der strukturierten Beschreibung anderer Modelle (letztere dienen der Abbildung von Systemen, Objekten oder einem Ausschnitt aus der Realität) [GeKP98]. Oft werden sie auch als „model of a modeling language“ [Seid03] bezeichnet. Ein Modell kann somit nur in Verbindung mit einem weiteren Modell als Metamodell betrachtet werden [Hars94], wodurch stets eine Kontextabhängigkeit gegeben ist. Die Vorgabe elementarer Modellierungskonstrukte hat das Ziel die verschiedenen Modellierungsprojekte zu vereinheitlichen. Auch im Kontext der Prozessmodellierung erfolgt über die Metamodelle die Definition der Prozessmodellierungssprachen, die dann zur Erstellung von Prozessmodellen verwendet werden.

Die Object Management Group (OMG) [OMG11c] hat einen Standard für die Verwaltung von Metamodellen veröffentlicht. Die sog. Meta Object Facility (MOF) [OMG11e] definiert ein Rahmenwerk zur Verwaltung von Metamodellen. [AkKü01], [AkKü05] und [HeGo06] haben dies entsprechend aufgenommen und weiterentwickelt (für eine kurze Beschreibung weiterer Ansätze sei auf [[KaKü02]] verwiesen). Es erfolgt die Definition einer mehrschichtigen, logischen Architektur, bei der jede Schicht eine bestimmte Art von Modellen aufnimmt. Die damit entstehende (Meta-) Modell-Hierarchie umfasst die Definition von Prozessmodellierungssprachen, Prozessmodellen sowie ausführbaren Instanzen dieser Prozessmodelle. Sie stellt dar, wie Anwender über die Definition von Prozessmodellierungssprachen und die Erstellung von Prozessmodellen letztendlich zu Prozessinstanzen gelangen (siehe dazu auch [KaKü02]).

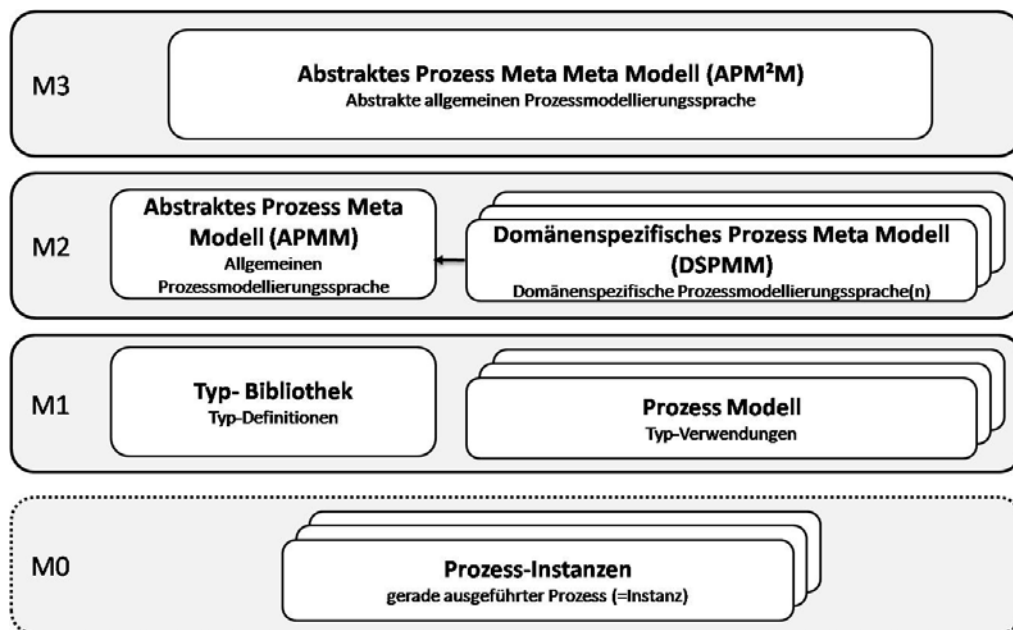


Abbildung 3-9 Meta-Modell-Hierarchie zu AOPM [JaVD09]

In Abbildung 3-9 ist eine Übersicht über die (Meta-) Modellhierarchie zu sehen; die Ebenen sind mit M3, M2, M1 und M0 bezeichnet. Jede der Ebenen ist als Instanz der höheren Ebene zu verstehen; die Ebenen sind also nicht unabhängig voneinander zu sehen.

Der einfacheren Darstellung wegen wird die Diskussion der Meta-Modell-Hierarchie nicht chronologisch gemäß Abbildung 3-9 durchgeführt, sondern sie beginnt mit Ebene M1. Es handelt sich dabei exemplarisch um die Meta-Model-Hierarchie zu AOPM [JaVD09], da AOPM bei den im Rahmen dieser Arbeit verwendeten Use Cases verwendet wurde und die entwickelten Konzepte darauf aufbauen.

M1- Prozess Modelle

Diese Ebene beinhaltet Prozessmodelle, welche für ein bestimmtes Anwendungsszenario in einer Organisation oder für die Konzeption einer Anwendung, definiert worden sind. Dazu kann prinzipiell jede beliebige Modellierungssprache verwendet werden. Für Beispiele bzgl. AOPM sei auf die Prozessabbildungen in Kapitel 2.2 verwiesen.

Gemäß dem sogenannten „Typ-Verwendungskonzept“ werden hier zunächst Prozesstypen definiert, die in eine Typ-Bibliothek gestellt werden. Neben Prozesstypen können auch Datentypen, Organisationstypen und weitere Bestandteile eines Prozessmodells definiert werden. Die verschiedenen Typen werden schließlich zur Erstellung von Prozessmodellen "verwendet", d.h. modelliert. Der Prozesstyp „Untersuchung durchführen“ beschreibt zum Beispiel allgemein wie ein Arzt eine Untersuchung durchzuführen hat, ohne dabei einen direkten Bezug zu einer Instanz, wie zum Beispiel Herrn Müller als Patient oder Dr. Mayer als behandelnden Arzt, zu haben.

Bei der Erstellung des Prozessmodells müssen außerdem Richtlinien für eine ordnungsgemäße Modellierung [BeRS95] [Rose96] eingehalten werden, um eine konsistente und korrekte Modellierung zu garantieren. Eine denkbare Regel für einen medizinischen Prozess wäre die Vorgabe, dass jedem Prozessschritt eine medizinische Rolle (Arzt, Krankenschwester, ...) zugeordnet werden muss.

Prinzipiell gilt, dass in den Prozessmodellen nur das abgebildet werden kann, was die Eigenschaften und Funktionen der verwendeten Modellierungssprachen beinhaltet.

M2 - Prozessmodellierungssprache

Bei der Definition der Prozesse auf Ebene M1 kommt eine entsprechende Prozessmodellierungssprache zum Einsatz. Während im Rahmen dieser Arbeit exemplarisch AOPM verwendet wird, könnte ebenso UML oder BPMN verwendet werden. Die Modellierungssprachen, inklusive aller Modellierungskonstrukte und dazugehörigen Eigenschaften sowie Modellierungsregeln, gilt es auf dieser Ebene zu definieren. M2 ist dazu in zwei Bereiche unterteilt.

- **Abstract Process Meta Model APM:** Zum einen wird eine Basissprache zur Verfügung gestellt, welche das grundlegende Modellierungskonzept wie zum Beispiel

Kontroll- oder Datenfluss definiert (siehe Abbildung 3-9, APMM). Die daraus resultierende allgemeine Prozessmodellierungssprache dient in einer Art abstrakten „Zwischensprache“ im zweiten Teil der Ebene als Basis für die Ableitung domänenspezifischer Prozessmodellierungssprachen. Erst diese können dann zur eigentlichen Modellierung auf M1 verwendet werden.

- **Domain Specific Meta Model DSPMM:** Im zweiten Teil werden die Konstrukte des APMM konkret definiert. Sie werden weiter angepasst oder auch gelöscht, wenn sie nicht essentiell sind. Ebenso können neue Konstrukte hinzugefügt werden. Die Bezeichnung als domänenspezifische Sprachen ist dabei nicht ganz exakt spezifiziert. Im Grunde sind damit allgemein auf M1 verwendbare Modellierungssprachen erfasst, die sich in zwei Gruppen einteilen lassen: *Standardmodellierungssprachen* wie BPMN oder UML sowie *anwendungsbezogene Modellierungssprachen*, wie sie mit AOPM definiert werden können (mit domänenspezifischen Sprachen sind im üblichen Sprachgebrauch die anwendungsbezogenen gemeint, so auch in der Gegenüberstellung in 3.3.2). So kann bei AOPM für den medizinischen Kontext zum Beispiel ein Konstrukt speziell für medizinische Entscheidungen, der sog. Evidenzbasierte Entscheider (näheres siehe [Meil05]), definiert werden. Das für eine Domäne angepasste Modell wird in einem eigenen Modell erfasst, in diesem Beispielfall im „Medical Domain Specific Process Meta Model“ MedDSPMM.

Im Rahmen dieser Arbeit gilt es den Blick der Anwender auch auf die anwendungsbezogenen Modellierungssprachen auszurichten.

M3 – Abstraktes Prozess Meta-Meta-Modell

Diese Ebene hat eine besondere Rolle. Sie stellt die höchste Abstraktionsstufe der Modellhierarchie dar und ist somit, wie [Petr06] es formuliert „self decriptive in that it is instance of itself“, d.h. es existiert kein übergeordnetes (Meta-) Modell. Das auf M3 definierte Modell, das *Abstract Process Meta Meta Model* APM²M, definiert nun eine Sprache, die dazu verwendet wird auf Ebene M2 Prozessmodellierungssprachen zu spezifizieren.

In dem verwendeten MOF Meta-Meta-Modell sind Modellierungskonstrukte und deren Typen, Beziehungen und Attribute enthalten. Es beinhaltet die Festlegung des grundlegenden Modellierungsparadigma, zum Beispiel als gerichteten, zyklischen Graphen, der sich aus Knoten und Kanten zusammensetzt.

M0 - Prozessinstanzen

Diese Ebene ist nicht mehr der Modellierungsumgebung, sondern der Ausführungsumgebung zu zuordnen. Sie enthält die Instanzen eines Prozesses zu dessen Ausführungszeit, beschreibt also die konkreten Aktivitäten, die in der Anwendung tatsächlich ausgeführt werden. So kann zum Beispiel aus dem Prozesstyp „Untersuchung durchführen“ (siehe M1) die Instanz „Untersuchung durchführen an Patient Herrn Müller durch Dr. Mayer als behandelnder Arzt“

abgeleitet werden. Die M0-Ebene wird von MOF nur der Vollständigkeit halber berücksichtigt. Im Rahmen dieser Arbeit wird sie Bestandteil des Gesamtkonzeptes sein, wie später in Kapitel 5 zu sehen sein wird.

Bewertung

Der Hauptvorteil dieser Modell-Hierarchie in Bezug auf den vorliegenden Kontext dieser Arbeit ist, dass es sich hierbei *nicht um eine starr vorgegebenes Prozessmodellierungssystem mit vordefinierter Prozessmodellierungssprache* handelt. Auf der Ebene M2 können beliebige Prozessmodellierungssprachen definiert werden entweder durch Anpassung einer vorhandenen oder durch eine komplette Neudefinition. So kann es vorkommen, dass bestimmte Sachverhalte mit bestehenden sprachlichen Mitteln zwar ausgedrückt, aber spezielle Informationen nicht angegeben werden können, zum Beispiel die Angabe der Durchlaufzeit eines Prozessschrittes. Im Rahmen der Meta-Modell-Hierarchie können solche zusätzlichen Eigenschaften auf M2 eingefügt werden - beispielsweise durch die Definition von zusätzlichen Attributen. Ist die Darstellung eines Aspektes mit den existierenden Elementen nur unzureichend möglich, kann die Einführung eines neuen Konstrukts auf M2 erfolgen. Der Evidenzbasierte Entscheider in [Meil05] bildet zum Beispiel eine Vielzahl medizinischer Einzelentscheidungen in einem Konstrukt ab. Im Gegensatz dazu können Konstrukte auch komplett aus einer Modellierungssprache gelöscht werden, wenn sie für den aktuellen Kontext nicht relevant sind. Ebene M3 soll in dem Zusammenhang nur der Vollständigkeit halber mit einbezogen werden. Es wird davon ausgegangen, dass die grundlegenden Modellierungsparadigmen weitestgehend unberührt bleiben; sie stellen die wesentlichen Grundpfeiler des Hierarchie-Modells dar.

Ein weiterer Vorteil der Meta-Modell-Hierarchie ist die *Strukturierung der Modellierung bzw. der resultierenden Modelle* gemäß der Modellebenen. Jeder Ebene kann ein Modelltyp mit bestimmten wesentlichen Eigenschaften zugeordnet werden: M3 ein Sprachdefinitionsmodell, M2 ein Sprachmodell, M1 die Prozessmodelle und M0 schließlich die ausführbaren/ausgeführten Prozessmodelle.

Nicht nur AOPM verwendet das Konzept der Metamodellierung. So basiert die UML seit der Version 2.0 (seit Mai 2010 ist 2.0 verabschiedet) auf dem Metamodell der MOF [OMG11e]. Zu den EPKs ist ein Meta-Modell in [BDKK02] zu finden. Bei BPMN ist in der Version 2.0 ein explizites Meta-Modell vorhanden [OMG11a].

3.3.4 Fazit zu den Grundlagen der Modellierung

Von den hier vorgestellten Modellierungssprachen ist lediglich AOPM individuell anpassbar. Die von UML und BPMN zur Verfügung gestellten Möglichkeiten sind unzureichend, da bei keiner der beiden Sprachen vollständig neue und unabhängige Aspekte eingefügt werden können und grundlegende Änderungen nur von verantwortlichen Organisationen

vorgenommen werden dürfen. Damit sollen (de-facto) Standards nicht als schlecht beurteilt werden. Wie in Kapitel 3.3.2.1 zu sehen war, haben sie auch ihre Vorteile. Die für die Standardmodellierungssprachen aufgezeigten Nachteile, sowie die der einzelnen Modellierungssprachen sollen vielmehr dazu ermutigen, bei entsprechend vorliegenden Kontextbedingungen die Definition einer anwendungsbezogenen Modellierungssprachen in Betracht zu ziehen, sollten die Nachteile bei einer Standardmodellierungssprache zu sehr überwiegen.

3.4 Fazit

Es kann Folgendes festgehalten werden: Das *Design* bildet die Grundlage bzw. Voraussetzung für eine gute Modellierbarkeit eines relevanten Anwendungsfalls mit individuellen Eigenschaften zu einem bestimmten Zweck. Dabei erfolgt zunächst eine genaue Identifikation und Analyse der Anforderungen. Dem schließt sich die Auswahl und/oder Gestaltung der entsprechenden Modellierungssprache inklusive der konzeptuellen Modellgestaltung [Mood05] [RIRG09] [Rose06a] an. Bei der *Modellierung* geht es „lediglich“ (der Vorgang ist in keiner Weise zu unterschätzen, wie es auch bei [MaSc97] und [PiRo95] zu lesen ist) um die Anwendung der Funktionen der Modellierungssprache zur Erstellung von Prozessmodellen [DaSh90] [DuAH05].

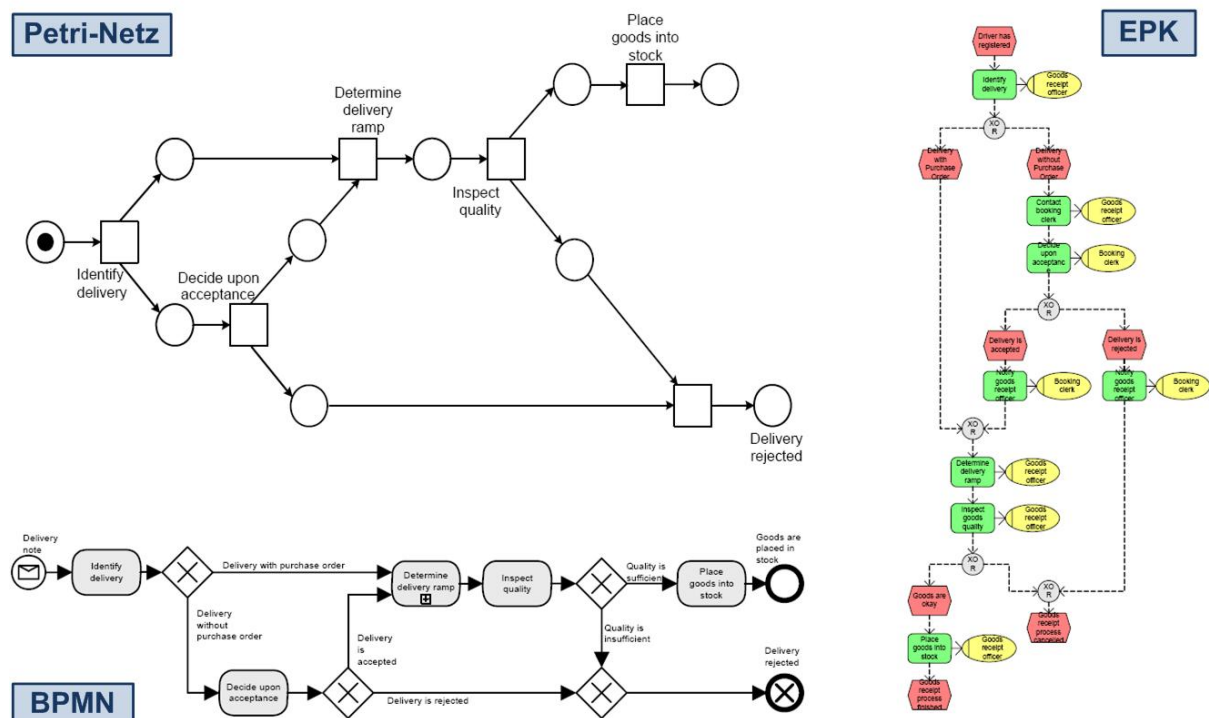


Abbildung 3-10 Beispiele Prozessmodellierungssprachen aus [Reck08]

Basierend auf den Analyseergebnissen aus Kapitel 3.2.3, den Studien und den eigenen Projekterfahrungen setzt die Kritik an den existierenden Ansätzen des Designs daran an, dass die Handlungsmöglichkeiten des Designs nur unzureichend genutzt werden und der Fokus zu sehr auf der eigentlichen Modellierung liegt. In der meist kaum explizit durchgeführten Designphase wird zu schnell eine bereits vorhandene Modellierungssprache bzw. Standardmodellierungssprache gewählt. Die Generierung einer individuellen, anwendungsbezogenen Sprache bleibt hingegen ungenutzt bzw. ist oftmals auch gar nicht möglich. Essentielle Aspekte in einem Prozessmodell können somit möglicherweise nicht erfasst werden. Zudem leidet die Produktivität der Modellierung dabei wesentlich [ClSW08]. Damit sollen die anwendungsbezogenen Sprachen nicht favorisiert werden. Vielmehr ist festzuhalten, dass im Hinblick auf die Verwendung einer Standardsprache oder anwendungsbezogenen Sprache objektiv, ohne einen konkreten Projekt- bzw. Anwendungskontext betrachtet, kein „Sprachtyp“ mit seinen jeweiligen Vorteilen überwiegt. Vielmehr ist eine kritische Beurteilung im Einzelfall notwendig [Mood05] [RIRG09] [Rose06a]. Die Vor- und Nachteile einer jeden Sprache können je nach Kontext unterschiedlich gewichtet werden. Als Gewichtungskriterien wären zuallererst der jeweilige Einsatzzweck und die dabei relevanten Aspekte zu nennen. Desweiteren ist das verfügbare Zeit- und/ oder Kostenbudget bzgl. der Entwicklung von Modellierungssprachen entscheidend, ebenso die Verfügbarkeit von entsprechenden Experten.

Abbildung 3-10 zeigt, dass es stets mehrere Möglichkeiten gibt ein- und denselben Ablauf darzustellen, da inzwischen eine Vielzahl von unterschiedlichen Modellierungssprachen existiert [AmSo07] [Sinu04] [SoWa07]. Hier sind ein Petri-Netz [Baum96], eine EPK (= Ereignisgesteuerte Prozessketten) [KeNS92] [Köhl06] [Mend08] sowie ein BPMN-Modell (= Business Process Modelling Notation) [OMG11a] zu sehen. Unterschiede sind beim Betrachten der Modelle leicht zu erkennen bzw. lassen sich bereits beim Vergleich der Modellierungssprachendefinition identifizieren. Folgende Unterschiede können beispielhaft genannt werden:

- *Unterschied (1):* Im EPK-Prozessmodell werden mit Ovalen Rollen definiert. In den anderen beiden Modellen hingegen fehlen derartige Angaben. Während dieser Aspekt beim BPMN-Modell bei der Modellierung schlicht weggelassen wurde, ist er für Petri-Netze gar nicht definiert.
- *Unterschied (2):* Während im BPMN-Modell die grauen Kästchen mit abgerundeten Ecken Aktivitäten darstellen, erfolgt bei den anderen beiden Prozessmodellen eine differenzierte, jedoch auch wieder unterschiedliche Abbildung. So werden beim EPK-Modell zwischen Funktionen und Ereignissen (Kästchen mit den runden Ecken und Sechsecke) unterschieden. Die Petri-Netze unterscheiden auf einem höheren Abstraktionsniveau zwischen Transitionen bzw. Aktivitäten und Stellen bzw. Plätzen (Quadrate und Kreise).

Die Beispiele zeigen, dass die Prozessmodellierungssprachen jeweils verschiedene Aspekte der Prozesse abbilden bzw. einzelne Aspekte in unterschiedlicher Art und Weise (siehe dazu auch [Reck08] und [SoWa07]). Die verschiedenen Darstellungsformen richten sich damit an unterschiedliche Zielgruppen, die Petri-Netze zum Beispiel an eher technisch-orientierte Mitarbeiter, BPMN auch an Mitarbeiter mit organisatorisch-betriebswirtschaftlichem Fokus [FrRH10]. Ob eine Modellierungssprache für einen Anwendungsfall adäquat ist oder nicht, ist somit relativ zu betrachten. Für eine erfolgreiche Modellierung ist es daher nicht nur entscheidend zu wissen, wie der eigentliche Prozessablauf inhaltlich aussieht. Diese Informationen liefert die konkrete Prozessaufnahme für die eigentlichen Modellierung (was in keiner Weise unterschätzt werden sollte). Vor allem aber müssen in der Designphase im Rahmen der Anforderungsanalyse folgende Fragen geklärt werden (siehe auch [Phal98], [AHMM02], [Reck08] und [Rose06b]):

- *Welche Art von Informationen* sollen überhaupt erfasst werden?
Während üblicherweise die Prozessschritte, Daten sowie Organisationen und Rollen erfasst werden, können für einzelne Unternehmen auch individuelle Kennzahlen relevant sein.
- *Welchem Zweck (welche Funktion)* haben die erfassten Informationen?
Wenn das Modell später für die automatisierte Ausführung verwendet wird, muss der dazugehörige Workflow syntaktisch wie auch semantisch eindeutig definiert sein. Bei graphischen Modellen ist die Zielgruppe der Leser entscheidend, je nachdem ob es der Manager ist, der nur einen groben Überblick über die Abläufe haben möchte, oder der Mitarbeiter auf der untersten Hierarchieebene des Unternehmens, der eine genaue Arbeitsanweisung benötigt.
- *Wie* sollen die Informationen erfasst werden [Mood05] [RIRG09] [Rose06a]
Die in diesem Kapitel vorgestellten Modellierungssprachen sind graphischer Art; es ist aber ebenso eine Abbildung als Text (sei es als reines Word-Dokument oder als Attribut eines graphischen Modellierungselements) möglich; ebenso gibt es tabellarische Beschreibungen. Auch kann sich der Detaillierungs- bzw. Abstraktionsgrad von Sprache zu Sprache unterscheiden.

Der Leser sollte sich bewusst sein, dass die Beantwortung dieser Fragen auch mit viel Erfahrung aus pragmatischen sowie praktischen Gründen eine nie vollständig bzw. perfekt zu lösenden Aufgabe ist [AHMM02]. Für Anforderungen, die von den gegebenen Prozessmodellierungssprachen nicht bzw. nicht adäquat umgesetzt werden können, für die Modellerstellung aber essentiell sind, muss jedoch die Möglichkeit bestehen, diese individuell in einer anwendungsbezogenen Prozessmodellierungssprache definieren zu können. Entscheidend ist es, diese Design-Aspekte vor den Start der eigentlichen Modellierung zu klären.

Um dieses Ziel zu erreichen, ist ein methodisches Rahmenwerk von Vorteil, das den Prozessdesigner bei seiner Aufgabe leitet. Die in Kapitel 3.3.3 vorgestellte Meta-Modell-

Hierarchie legt den Fokus zunächst einmal auf die Modellerstellung. Darauf aufbauend kann eine klare Spezifikation der Aufgaben von Design und Modellierung erfolgen. Die Meta-Modell-Hierarchie liefert zudem mit ihren einzelnen Ebenen Strukturierungskriterien für verschiedene Modellarten. Sie bietet somit die Möglichkeit, die Aufgaben des Designs sowie die ermittelte Anforderungen in Anlehnung an die verschiedenen Ebenen der Modellierung zu strukturieren. Da in den aktuellen Design-Publikationen oftmals eine fehlende Methodik kritisiert wird (siehe hierzu zum Beispiel die Untersuchungen von [InRe08]) soll die Meta-Modell-Hierarchie als methodische Grundlage für das im Rahmen dieser Arbeit ausgearbeitete Design-Konzept dienen. Die Erläuterungen dazu folgen in Kapitel 5, während zuvor in Kapitel 4 konkrete Anwendungsfälle erläutert werden, um die Relevanz des Designs zu unterstreichen.

Kapitel 4

Design und Modellierung – Anforderungen

Die Spezifikation der Anforderungen im Rahmen des Designs stellt einen ebenso wichtigen wie auch schwierigen Schritt dar [DaSh90] [Ehrl07] [VeCo94]. Dabei kann es um die Entwicklung eines Informationssystems gehen [VeCo94] oder um die Herstellung eines physischen Produktes in der Produktentwicklung [Ehrl07] [PaBe97]. Auf Basis der Anforderungen wird ein Modell des zu gestaltenden Objektes entwickelt, das dann in die Realität umgesetzt wird. Dies kann die informationstechnisch unterstützte Ausführung eines Prozessmodells oder, im Bezug auf die Produktentwicklung, ein konkretes Produkt, wie zum Beispiel ein geschliffenes Metallteil mit bestimmten geometrischen Abmessungen beinhalten.

In diesem Kapitel werden Problemfälle und Anforderungen aus drei (Modellierungs-) Projekten des Lehrstuhls für Angewandte Informatik IV an der Universität Bayreuth erläutert. Es wird gezeigt, wie die Anforderungen derzeit umgesetzt werden; dazu werden beispielhaft die in Kapitel 3.3.1 eingeführten Modellierungssprachen herangezogen. Darüber hinaus wird aufgezeigt welche Verbesserungsmöglichkeiten es geben könnte.

4.1 Übersicht über die Projekte

Um die späteren Ausführungen besser verstehen zu können, wird mit einer kurzen Beschreibung der Projekte begonnen.

- **Projekte 1 – Produktentwicklungsprozesse**

Im Rahmen des von der Bayerischen Forschungsförderung geförderten Forschungsverbunds „FORFLOW“ ([MePa06], [MePa08]) galt es einen sog. Prozessnavigator zur Prozess- und Workflowunterstützung von Produktentwicklungsprozessen zu erforschen, zu entwickeln und prototypisch umzusetzen.

Die Grundlage für die Unterstützung bildete die Generierung eines umfassenden Prozessmodells, das alle für den Produktentwickler notwendigen Informationen abbilden sollte. Als Grundstruktur wurden dabei das V-Modell@XT [Auto06] verwendet, ergänzt um wesentliche Aspekte der VDI-Richtlinien [VDI04] [VDI82] [VDI97] sowie das Vorgehensmodell nach Pahl/Beitz [PaBe97]. Mit der aspektorientierten Prozessmodellierung inklusive dem Modellierungswerkzeug i>pm (Kapitel 3.3.1.4) stand bereits ein flexibles Modellierungsframework zu Verfügung. Dabei mussten jedoch noch zusätzliche Eigenschaften, die sich aus dem Projekt

ergaben, in die Modellansicht und damit letztendlich auch in den Prozessnavigator, integriert werden. Dies waren zum einen die für die Produktentwicklung charakteristischen Phasen, durch die der Gesamtprozess in überschaubare Teilabschnitte unterteilt werden kann. Eine weitere Anforderung bestand darin, Informationen über methodische Vorgehensweisen bei der Entwicklungsarbeit geben zu können [Meer07b]. Das Gesamtmodell umfasst ca. 100 Schritte. Eine ausführliche und komplette Darstellung des sog. FORFLOW-Prozessmodells findet sich in [KELR10].

- **Projekt 2 – Medizinische Prozesse**

Im Rahmen eines Qualitätsmanagementprojektes zur Einführung Klinischer Behandlungspfade am Klinikum in Fürth erfolgte eine Ist-Analyse der im Klinikum vorhandenen medizinischen sowie administrativen Prozesse. Daran schloss sich deren Dokumentation in einem Prozessmodell an [FMSJ09]. Ziel war die Erstellung einer patientenorientierten Prozesslandschaft bestehend aus den Kernprozessen des Klinikums gemäß den DRGs (DRG = Diagnosis Related Groups, dt. Diagnosebezogene Fallgruppen, Klassifikationssystem für Leistungen am Patienten [FeBD90] [Fisc01]). Folgende DRGs wurden aufgenommen: Hüft-TEP, Koronare Herzkrankheit, Gastroenteritis, Gallensteinleiden, Leistenhernie und Schädelprellung. Von den unterstützenden Prozessen wurden Pflegeüberleitung, operative Intensivpflege, der Sozialdienst und die Klinikums-Apotheke erfasst. Jedes Teilmodell umfasst ca. 150 – 200 Schritte.

Es galt die Abläufe inklusive der Verantwortlichen, der notwendigen medizinischen Geräte (Werkzeug), Dokumente bzw. Informationen (bzw. allgemein der In- bzw. Output der Schritte zum Beispiel in Form eines Medikaments) zu erfassen. Es sollte die Dokumentation nicht bis ins letzte Detail oder mit allen Sonderfällen erfolgen; es galt vielmehr eine Art Richtschnur bzw. Standard für die Prozesse abzubilden. Wichtig war es, die Schnittstellen zwischen den Kernprozessen und den unterstützenden Prozessen zu identifizieren.

- **Projekt 3 – Verwaltungs- bzw. universitätsbezogene Prozesse**

Dieses Projekt fand im Rahmen einer Qualitätsmanagementmaßnahme der Universität Bayreuth statt. Im Hinblick auf die von der Universitätsleitung angestrebte Systemakkreditierung, galt es für die zuständige Akkreditierungsagentur eine Dokumentation der Universität zu dem Bereich Studium und Lehre zu erstellen [UnBa11]. Dazu wurden unter anderem die Prozesse aus dem Bereich Studium und Lehre (Einführung, Durchführung = „Student-LifeCycle“, Aufhebung eines Studiengangs) in einem Prozessmodell abgebildet und umfassend in einem dazugehörigen Prozesshandbuch erläutert. Für eine transparente Darstellung galt es vor allem die Verantwortlichen und alle Beteiligten inklusive ihrer Interaktion und Kommunikation sowie die relevanten Dokumente, Informationen und Zeitangaben zu spezifizieren. Das Gesamtmodell umfasst ca. 250 Schritte.

Die Modellierung erfolgte in allen drei Projekten unter Verwendung der aspektorientierten Modellierungssprache AOPM und dem Modellierungswerkzeug i>pm (siehe Kapitel 3.3.1.4).

4.2 Anforderungen

Im folgenden Kapitel wird nun auf ausgewählte Anforderungen aus den in Kapitel 4.1 erläuterten Projekten eingegangen. Ein Anspruch auf Vollständigkeit wird dabei nicht erhoben. Die Auswahl erfolgte gemäß Relevanz der einzelnen Anforderungen in den verschiedenen Modellierungsprojekten, d.h. inwiefern sie in mehr als einem Projekt identifiziert worden sind.

4.2.1 Phasen

Projekte werden für ihre Durchführung meist in Phasen unterteilt. Nach DIN 69901 ist eine Projektphase ein „zeitlicher Abschnitt eines Projektverlaufs, der sachlich gegenüber anderen Abschnitten getrennt ist“ [DeIn09]. Beispiele sind in Tabelle 4-1 zu sehen.

	Beschreibung
Beispiel (1)	Ein Produktentwicklungsprozess wird üblicherweise in die Phasen Planen, Konzipieren, Entwerfen und Ausarbeiten unterteilt. Diese lassen sich jeweils in weitere Arbeitsschritte unterteilen [Ehrl07].
Beispiel (2)	Der Verlauf eines Studiengangs an einer Universität kann in die Phasen der Planung, Einrichtung, der Durchführung sowie der Aufhebung gegliedert werden (siehe zum Beispiel [UnBa11]). Sie lassen sich mit einer Vielzahl von Arbeitsschritten konkretisieren.

Tabelle 4-1 Beispiele Phasen

Anforderungen

Phasen werden nicht als konkrete Aufgabe gesehen wie es bei Prozessen der Fall ist, sondern besitzen einen *aufgaben- bzw. prozessübergreifenden Charakter* [Birk06] [VDI93]. Daher erfolgt die Definition der Phasen sehr grob, jedoch als logisch abgeschlossene Einheit [Burg02]. Zur Konkretisierung erfolgt eine *Dekomposition der Phasen in Prozesse* [Birk06] [VDI93]. Die Phasen sollen sich aufgrund des abweichenden Charakters in der graphischen Darstellung von Prozessen unterscheiden.

Die Phasen unterscheiden sich zudem in ihrem *zeitlichen Aufwand* [Burg02] [Ehrl07]. Daher ist die Angabe eines Zeitfaktors wünschenswert, um das Gesamtbild besser darstellen zu können.

Auch wenn es per Definition (siehe zum Beispiel [Burg02]) nicht vorgesehen ist, hat die Erfahrung gezeigt, dass sich die Phasen oftmals *zeitlich überlappen*. Es findet kein sequentielles Hintereinanderschalten der Phasen statt, wie es bei Prozessen mittels Kontroll- und Datenfluss abgebildet wird; vielmehr findet teils eine parallele Ausführung statt. So kann mit dem Verkauf der ersten Produkte begonnen werden, während die Produktion noch weiter läuft.

Rücksprünge in frühere Phasen sind zum Zweck der Ergebnisoptimierung essentiell. Ziel der Modellierung kann es jedoch nicht sein, alle denkbaren Rücksprünge im Voraus zu identifizieren, um diese dann alle abzubilden. Dies würde zu unübersichtlich werden und wäre auch sehr aufwendig.

Phasen enden üblicherweise mit sog. *Meilensteinen*, die ein zu erzielendes Ergebnis (zum Beispiel den Produktentwurf) definieren [Birk06] [Wild07]. Im Unterschied zu den Output-Daten eines Prozesses sind die Ergebnisse im Bezug auf einen Meilenstein von strategischer Bedeutung [DeIn09] und dienen der übergeordneten Strukturierung der Abläufe [Wild07]. Sie sollten daher auch bei der Modellierung hervorgehoben werden.

Es ist zudem sinnvoll auf Ebene der Phasen *Rollen* zu definieren, um Verantwortliche spezifizieren zu können.

Werkzeuge oder Systeme gilt es erst für die Prozessschritte anzugeben, da sie im Bezug zu einer konkreten Tätigkeit gesehen werden.

Lösungsansätze

EPKs oder **UML-Diagramme**, wie das Aktivitäts-oder das Anwendungsfalldiagramm, bieten kein Modellierungskonstrukt zur expliziten Darstellung von Phasen. Hier liegt der Schwerpunkt auf der Darstellung von Prozessen zur Detailbeschreibung.

Der Standard **BPMN** definiert die Konstrukte Lanes sowie Pools sehr ungenau; diese könnten daher ersatzweise zur Darstellung des prozessübergreifenden Charakters der Phasen verwendet werden. Da dies jedoch leicht zu weiteren Missverständnissen führt, sollte davon Abstand genommen werden. Zudem lassen sich die gewünschten zeitlichen Aspekte damit auch nicht darstellen. Die Verwendung von Aktivitäten ermöglicht keine Abgrenzung zu den „normalen“ Prozessen. Das Artefakt zur Gruppierung kann für die Zusammenfassung jeglicher Art von Elementen verwendet werden, so dass die Eindeutigkeit sowie Klarheit in der Darstellung nicht gegeben ist. Eine Ausarbeitung eines individuellen Symbols für die Darstellung der Phase wäre mit BPMN 2.0 möglich, doch wäre das Konstrukt dann nicht Teil des Standards.

Für **AOPM** wurde in einer speziellen, für die Produktentwicklung angepassten Version, das Modellierungskonstrukt **PEP** (= *ProduktEntwicklungsPhase*) definiert. Dies dient der *übergreifenden Zusammenfassung von Prozessen*. Die Phasen stellen Superprozesse zu den

bisher bekannten Prozessen dar, d.h. sie bilden die oberste Ebene eines Prozessmodells und auf der darunterliegenden Ebene werden die Prozesse modelliert. Die Anwendung erfolgt inzwischen nicht nur bei der Modellierung von Produktentwicklungsprozessen, sondern domänenunabhängig, wie es in Abbildung 4-1 an Hand eines Beispiels aus der Medizin gezeigt wird.

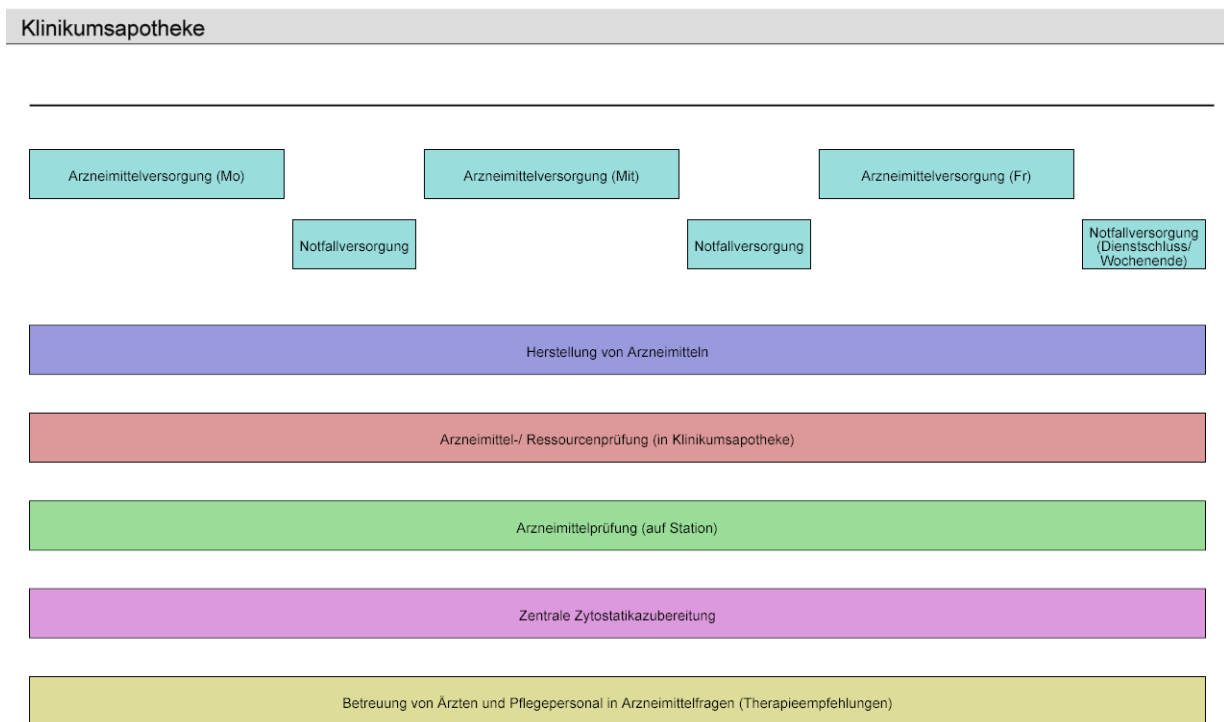


Abbildung 4-1 Phasen mit AOPM im i>pm modelliert

Die graphische Darstellung der Phasen erfolgt in Form von Balken, die untereinander angeordnet werden. Ihre Länge und Position kann in horizontaler Richtung individuell zur Darstellung des *Zeitfaktors* bestimmt werden; eine Überlappung der Phasen ist somit *zeitlich und inhaltlich* möglich. Eine explizite Darstellung von Kontroll- bzw. Datenflüsse erfolgt nicht.

Der *organisatorische und operationale Aspekt* wird bei der im Rahmen dieser Arbeit verwendeten Version des i>pm erst auf der darunter liegenden Ebene modelliert. Sinnvoll wäre es aber, zumindest auch den organisatorischen Aspekt schon auf Ebene der Phasen verwenden zu können.

Meilensteine können prinzipiell auf Phasen- und Prozessebene modelliert werden. Werden sie auf beiden Ebenen modelliert, ist damit eine Unterscheidung in (Zwischen-) Meilensteine zur Erfassung von Zwischenergebnissen auf Prozessebene und Meilensteine zur Erfassung von Ergebnissen einer Phase möglich. Zum Zweck der Einheitlichkeit und Übersichtlichkeit

erscheint es sinnvoll, Meilensteine nur auf einer der beiden Ebene zu definieren. Die Definition auf Prozessebene ist dabei von Vorteil da sowohl detaillierte Zwischenmeilensteine als auch Meilensteine in Bezug auf das Ende einer Phase (siehe Abbildung 4-2)⁵ definiert werden können. Die Meilensteine werden in dem Fall über den Kontroll- oder Datenfluss an einer beliebigen Stelle in den Prozessablauf integriert. Zusätzlich besteht die Möglichkeit, Daten, d.h. Dokumente, an diesem Konstrukt zu hinterlegen, was hier nicht dargestellt ist. Werden die Meilensteine nur auf Phasenebene modelliert, gehen die Informationen bzgl. der Zwischenmeilensteine verloren.

AOPM liefert somit explizit ein Konzept zur Repräsentation von Phasen das die ermittelten Anforderungen abdecken kann.

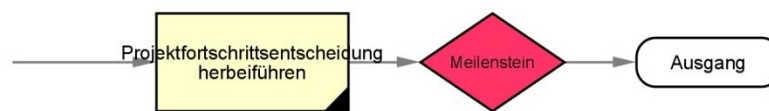


Abbildung 4-2 Meilenstein mit AOPM im i>pm modelliert

4.2.2 Kreative bzw. „unscharfe“ Prozesse

Als kreative Prozesse werden solche Prozesse bezeichnet, die eine Aufgabe mit einem hohen Maß an Einfallsreichtum lösen und schließlich ein reales oder logisch fassbares Ergebnis erzielen [Laut11]. Daneben gibt es Prozesse, die nicht zwingend Kreativität erfordern, sondern sich durch ein Höchstmaß an Abhängigkeiten bzgl. der bei der Ausführung vorliegenden Bedingungen auszeichnen. Diese sind entscheidend für den realen Prozessablauf und bestimmen welche Schritte notwendig sind. Beispiele sind in Tabelle 4-2 zu sehen.

	Beschreibung
Beispiel (1)	Die Planung der universitären Lehrveranstaltungen bzgl. Raum- und Zeitplanung ist ein Prozess mit vielen Iterationen und oftmals sind viele zeitliche oder räumliche Verschiebungen notwendig, bis das Endergebnis feststeht.
Beispiel (2)	Die Behandlung eines Patienten (vor allem bei Notfällen) kann nie exakt im Voraus in einem Prozessmodell abgebildet werden, weil stets auf den aktuellen Gesundheitszustand des Patienten reagiert werden muss.

Tabelle 4-2 Beispiele kreativer bzw. unscharfer Prozesse

⁵ Aus implementierungstechnischen Gründen kann bei Verwendung von i>pm der Schriftzug zur Benennung des Prozesses nicht an die geometrischen Abmessungen des Prozessbausteins angepasst werden.

Anforderungen

Kreative bzw. „unscharfe“ Prozesse lassen sich bzgl. eines genauen Ablaufs sowohl inhaltlich als auch strukturell schwer konkretisieren. Der *Formalisierungsgrad* der Schritte ist sehr *gering*. Die kreativen Ideen entstehen dabei hauptsächlich im Kopf der ausführenden Person während der eigentlichen Ausführung, deren explizite Abbildung zum Zeitpunkt der Modellierung verständlicherweise nicht möglich ist [Pete04]. Sie weisen nur wenige, standardisierte bzw. obligatorische Tätigkeiten auf [Hofe99] [PaBe97]. Da die Gesamtanzahl der Schritte an sich jedoch sehr hoch ist, macht es Sinn sie zu *kompositen Schritten* zusammenzufassen [Hofe99] [PaBe97].

Der Grad der *Vernetzung* zwischen den einzelnen kreativen Teilschritten ist relativ hoch [Hofe99]. Zudem werden *viele Iterationsschleifen* durchlaufen, d.h. es erfolgt keine systematische Abhandlung der Aktivitäten, weil die Reihenfolge zur Modellierungszeit nicht genau vorgegeben werden kann [Pete04].

Mit dem Fortschritt der Durchführung der kreativen bzw. unscharfen Prozesse ist eine *zunehmende Informationsschärfe*, quantitativ wie qualitativ, zu verzeichnen [Hofe99]. Der funktionale Aspekt steht dabei weniger im Vordergrund als der zu erzielende Output. Dieser lässt sich zum Zeitpunkt der Modellierung zumindest als *abstraktes Ergebnis* in Form von zu erstellenden Dokumenten oder Ähnlichem definieren.

Werkzeuge, vor allem IT-Systeme, dienen der Unterstützung bei der Dokumentation bzw. nachfolgenden Umsetzung der kreativen Gedanken und sollten daher abgebildet werden können.

Die Darstellung eines kreativen bzw. „unscharfen“ Prozesses sollte sich von einem „normalen“ Prozess nicht zu grundlegend unterscheiden, wie es hingegen bei den Phasen gefordert wurde. Der Charakter eines Prozesses an sich ist immer noch vorhanden. Zudem sind sie Bestandteil des Prozessablaufs; eine *Einbindung in den Kontroll- bzw. Datenfluss* muss also möglich sein.

Lösungsansätze

Die von der **UML** hauptsächlich zur Prozessmodellierung verwendeten Aktivitätsdiagramme definieren im Sinne eines funktionalen Aspektes die Aktivitäten und ihre Vernetzungen. Bei den kreativen bzw. „unscharfen“ Prozessen jedoch können weder alle Aktivitäten genau spezifiziert werden, noch kann ihre Vernetzung genau angegeben werden. Eine explizite Darstellung einer anderen Prozessart, wie die eines kreativen Schrittes, ist nicht möglich.

EPKs sind von ihrem Grundgedanken her bereits ungeeignet für diese Art von Prozessen, da sie den Fokus auf die Modellierung standardisierter Abläufe legen. Bei diesen sind die Ereignisse und Funktionen sowie deren strukturellen Zusammenhänge definiert, so wie sie allgemein anerkannt sind und angewandt werden.

Auch **BPMN** legt den Schwerpunkt auf die Darstellung der zeitlogischen Abfolge von Aktivitäten. Bei den kreativen Prozessen soll zwar ein logisch nachvollziehbares Ergebnis erzielt werden, doch spiegelt sich das nicht immer in dem davor ablaufenden Prozess wieder. Eine Abbildung ist mit den vorhandenen (funktionalen) Modellierungselementen somit nicht möglich.

Bei **AOPM** wird bzgl. des funktionalen Aspektes derzeit nur zwischen den Phasen sowie (kompositen oder elementaren) Prozessen unterschieden. Im Gegensatz zu den vorher genannten Modellierungssprachen wäre die Einführung eines entsprechenden Konstruktes jedoch jederzeit möglich. Dieses sollte graphisch einem normalen Prozessschritt ähnlich sein und als kompositen Element definiert werden. Im Hinblick auf die Subprozesse muss es möglich sein die obligatorischen Schritte und, wenn bekannt, auch ihre Ablaufreihenfolge zu definieren. Die Angabe von Input und Output sollte ebenfalls möglich sein.

Das Konzept des Platzhalters (siehe zum Beispiel [Hofe99]) ist in keinem der uns bekannten (Standard-) Modellierungssprachen vertreten. In vielen Fällen, unabhängig von den erläuterten Modellierungssprachen, hat sich jedoch die Modellierung von *Platzhaltern* für diese Art von Prozessen als geeignet bzw. ausreichend erwiesen [Hofe99]. Sie stellen eine *abstrakte Funktion* im Sinne einer „Black Box“ dar. Die Definition weiterer Unterschritte ist nicht im Konzept vorgesehen, somit ist die Angabe obligatorischer Teilschritte nicht möglich; Zyklen und Rücksprünge sind nicht berücksichtigt. Darüber hinaus werden keine Angaben zur graphischen Darstellung gemacht. Die zur Modellierungszeit bekannten *Rahmenbedingungen* können wie bei einem normalen Schritt als Input angegeben werden, ebenso die *abstrakt formulierten Ergebnisse*. Durch die fehlende Dekompositionsmöglichkeit ist eine Angabe von Zwischenergebnissen nicht möglich.

Zusammenfassend kann festgehalten werden, dass das Konzept der Platzhalter noch verbesserungswürdig ist, jedoch einen Ansatz in die richtige Richtung darstellt. Im AOPM wäre dies dann entsprechend zeitnah umsetzbar.

4.2.3 Umfangreiche Prozessmodelle

Bei großen Organisationen, die in ihrem Portfolio breit aufgestellt sind, aber auch in einer Klinik oder Universität, werden viele verschiedene Tätigkeiten ausgeführt, bei denen es eine Vielzahl weiterer Aspekte zu berücksichtigen gilt. Bei der Modellierung, wie auch bei der Modellanwendung, kann leicht der Überblick verlorengehen. Beispiele sind in Tabelle 4-3 zu sehen.

	Beschreibung
Beispiel (1)	Große Menge an Informationen, Dokumenten, Gesetzen, Beschlüssen, die im Rahmen der Einrichtung eines Studiengangs an einer Universität relevant sind. Die dazugehörigen Schritte werden außerdem von den unterschiedlichsten

	Rollen der Universität (aus der Verwaltung sowie den Fakultäten) durchgeführt.
Beispiel (2)	An der Durchführung einer Hüft-TEP sind, angefangen bei der prä-stationären Sprechstunde, bis hin zur Entlassung mit Nachsorge, viele verschiedene Personen des administrativen und vor allem des medizinischen Personals beteiligt. Für die verschiedenen Untersuchungen, die Operation sowie die Rehabilitation werden die unterschiedlichsten Geräte benötigt.

Tabelle 4-3 Beispiele umfangreicher Prozesse

Anforderungen

Zum besseren Verständnis beim Lesen der Prozessmodelle sollte die *große Menge an Elementen logisch strukturiert* werden. Dies sollte für jeden Aspekt des jeweiligen Prozessmodells beachtet werden, vor allem für den funktionalen Aspekt. Sieben Elemente auf einem Prozessblatt, die Anzahl, die ein (Modell-) Leser maximal auf einen Blick erfassen kann (siehe dazu [Mill56]), sind meist schnell erreicht. Die einzelnen *Schritte* müssen daher in logischen Einheiten, die inhaltlich abgegrenzt sind, organisiert werden können. Oftmals sind Prozesse auch von einer hohen Anzahl an *Entscheidungen* geprägt, wodurch sich die strukturelle Komplexität der Prozesse zusätzlich erhöht. Dabei soll sich der Anwender nicht in der Vielzahl der Entscheidungen verlieren. Vielmehr sollten diese bei ihrer Abbildung sinnvoll kombiniert werden. Mit den vielen Prozessschritten wird zudem eine große Menge an Daten verarbeitet sowie produziert. Die Darstellung jedes einzelnen Datenelements lässt die Modelle schnell unübersichtlich und überladen aussehen. Eine entsprechende Abstraktion ist daher notwendig.

In einigen Fällen kann auch die Menge der einem Schritt zugewiesenen *Rollen und/ oder Werkzeuge* eine zusätzliche Ordnungsstruktur bzw. Verwaltungskonzept erforderlich machen (siehe für organisatorischen Aspekt zum Beispiel [Buss98]). Werden einem einzelnen Prozess mehrere Rollen und mehrere Werkzeuge zugeordnet, muss klar sein, wie diese zueinander in *Beziehung* stehen und wie sie mit einander kombiniert werden dürfen. Selbiges muss außerdem zwischen den einzelnen Prozessen definiert werden.

Der Output eines Prozessschrittes wird nicht immer komplett zu dem direkt nachfolgenden Schritt geleitet, sondern verteilt sich auf mehrere nachfolgende Schritte. Umgekehrt kann der Input eines Schrittes aus unterschiedlichen Quellen stammen. Ziel muss es sein diese vielen *unterschiedlichen Out- und Input-Beziehungen* übersichtlich und vollständig in den Prozessmodellen abzubilden.

Die Darstellung der Abhängigkeit von Prozesselementen gilt sowohl *zwischen den Aspekten* als auch auf den einzelnen Aspekt an sich bezogen, wenn nur ein Prozessschritt für sich oder aber der Gesamtprozess betrachtet wird. Die Rolle an Prozessschritt A kann die Wahl der

Rolle oder des Werkzeuges an einem nachfolgenden Schritt beeinflussen. Verkompliziert wird die Sache noch weitaus mehr, wenn nicht nur die Kombination Rolle und Werkzeug betrachtet wird, sondern weitere Aspekte mit einbezogen werden. Eine verständliche und nachvollziehbare Darstellung ist dabei essentiell. Spätestens bei der Ausführung der Prozesse muss ein entsprechender Mechanismus integriert sein.

Lösungsansätze

Ein verbreiteter Ansatz zur Beherrschung der Komplexität umfangreicher Prozessmodelle ist das Konzept der *kompositen Modellierung von Elementen*. Komposite Prozesse fassen logisch abgrenzbare Abschnitte eines Prozesses auf einer höheren Ebene zusammen, so dass sich das Modell in einzelne Hierarchieebenen untergliedern lässt. Ein Beispiel ist in Abbildung 4-3 zu sehen, in der die kompositen Prozesse an den schwarzen Ecken zu erkennen sind. Sowohl die komposite als auch die dekomposite Ebene weisen ihren eigenen Detaillierungs- bzw. Abstraktionsgrad auf.

Das Konzept der (De-) Komposition lässt sich auch auf die weiteren Aspekte der Prozessmodelle übertragen. Es können komposite Daten im Sinne von sog. Arbeitsmappen zur logischen Zusammenfassung von Dokumenten definiert werden. In Abbildung 4-3⁶ ist dies bereits realisiert, zu erkennen an der schwarzen Ecke der modellierten Daten. Wie bei den Prozessen sind auf der darunterliegenden Ebene die detaillierten Daten modelliert. Ebenso können komposite Organisationseinheiten erstellt werden, wodurch die Definition von Teams, Arbeitsgruppen oder Abteilungen möglich wird; gleiches gilt für die Werkzeuge. Bei allen Aspekten wird damit eine hierarchische Modellierung der Elemente vorgenommen.

Entscheidungen lassen sich ebenfalls zu einem kompositen Konstrukt zusammenfassen, so dass die Einzelentscheidungen abstrahiert als eine übergeordnete Gesamtentscheidung dargestellt werden können. Näheres bzgl. Lösungsansätze folgt separat in Kapitel 4.2.5.

Speziell im Bezug auf den funktionalen Aspekt ist zur weiteren Komplexitätsbeherrschung die Erstellung von *Teilmodellen* möglich. Diese werden über Schnittstellen zu einem Gesamtmodell zusammengeführt; das Gesamtmodell bildet den obersten kompositen Prozess in der Hierarchie. Es kann ebenso sein, dass die Teilmodelle nicht einzelne Abschnitte des Hauptkontrollflusses des Gesamtmodells darstellen, sondern produkt- bzw. objektorientiert definiert sind. Dann müssen Prozessabschnitte redundant modelliert werden, wozu das Typ-Verwendungskonzept verwendet werden kann (siehe Kapitel 3.3.1.4). So wurde im Rahmen des medizinischen Projektes (siehe Kapitel 4) für jede der fünf TOP-DRG ein Prozessmodell erstellt. Dabei beinhalten zum Beispiel sowohl der Prozess zur Hüft-TEP als auch der zur Koronaren Herzkrankheit den klinischen Aufnahmeprozess.

⁶ Es besteht bei dieser Abbildung nicht der Anspruch, dass der Leser alle Details des Prozessmodells lesen kann; der Fokus liegt auf der prinzipiellen Darstellung der kompositen Modellierung.

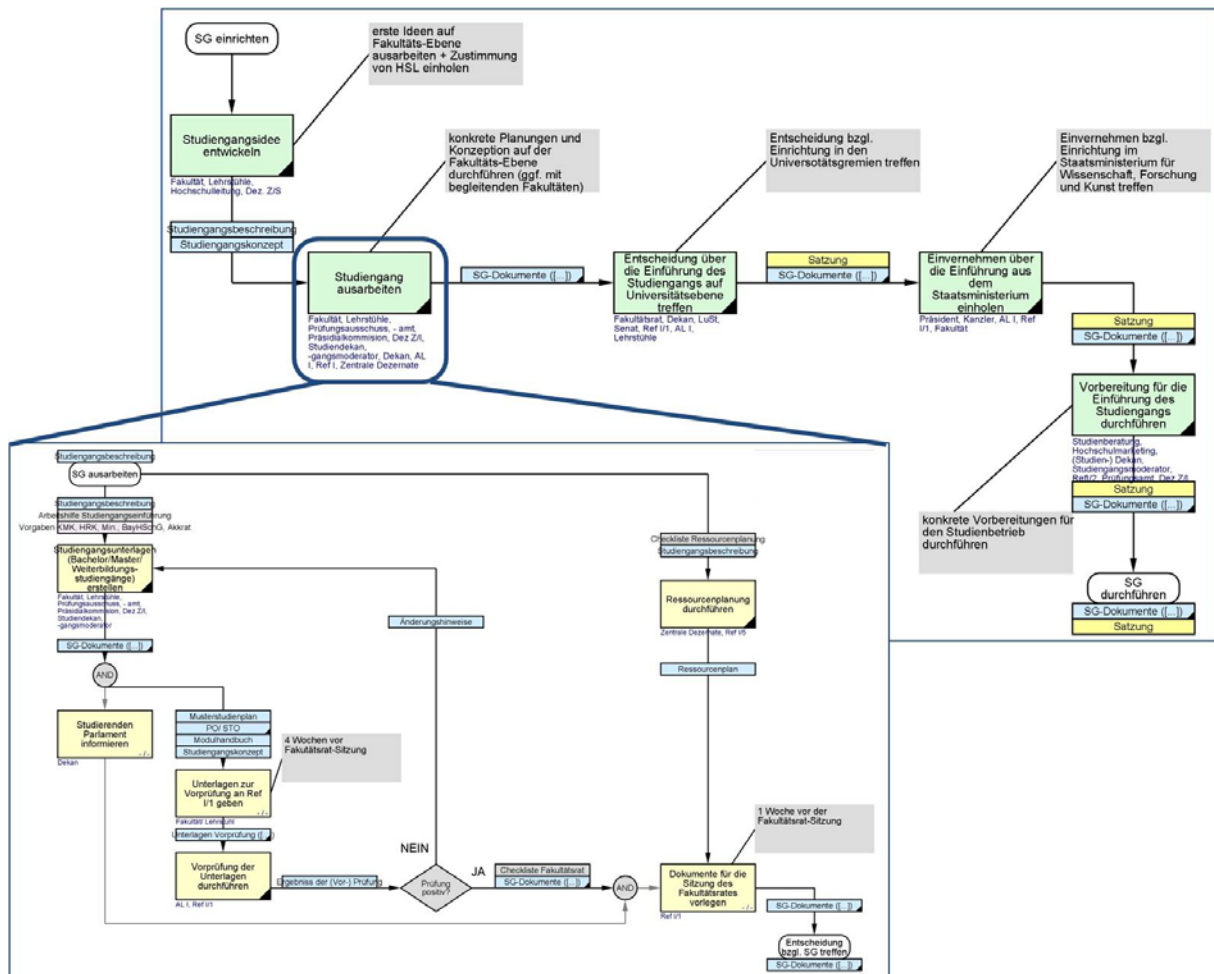


Abbildung 4-3 Beispiel komposite Modellierung

Bei den Aktivitätsdiagrammen der **UML** ist eine komposite Modellierung von Aktivitäten möglich; die Dekomposition ist bei den übrigen Modellierungselementen nicht vorgesehen. Aktivitätsdiagramme können andere Aktivitätsdiagramme oder Teile davon referenzieren, so dass eine Teilmodellbildung möglich ist.

Bei **EPKs** ist mit der sog. *Prozessverfeinerung* ein Konzept zur (De-) Komposition vorgesehen. Es gibt zudem sog. *Prozesswegweiser*, die zum Verweis auf andere Prozesse verwendet werden können. Durch die Angabe einer Referenz, setzen sie mindestens zwei verschiedene Prozesse miteinander in Beziehung. Somit kann ein Gesamtmodell in Teilmodelle zerlegt werden. Die Modellierung umfangreicher Prozesse wird jedoch generell als problematisch angesehen.

BPMN verfügt in der neuen Version 2.0 neben kompositen Prozessen auch über sog. *Listen-Datenobjekte* zur Repräsentation einer Gruppe von Informationen. Die komposite Darstellung von Rollen oder Werkzeugen ist bei BPMN im Ansatz jedoch nur unzureichend möglich. Zwar können sog. Pools zur Zusammenfassung von Elementen verwendet werden und Lanes

zur weiteren Unterteilung. Jedoch sind diese Modellierungselemente nicht eindeutig hinsichtlich der Bedeutung definiert (siehe dazu zum Beispiel [Reck10]). Das Konstrukt *Gruppierung* ermöglicht zwar ein Zusammenfassen von Prozesselementen, doch wird das komposite Modellierungskonstrukt sowie die dazugehörigen dekompositen Elemente (graphisch) auf einer Ebene dargestellt, so dass keine Informationen zum Zweck der Komplexitätsbeherrschung verborgen werden können.

Die Erstellung von Teilmodellen und deren Abbildung in Form sog. Prozesslandschaften, um einen Überblick über die vorhandenen Prozesse und ihre Zusammenhänge zu geben, ist mit BPMN explizit nicht möglich. Der Fokus liegt klar auf der Darstellung zeitlich-logischer Abfolgen von Aktivitäten (siehe dazu auch [FrRH10]).

AOPM realisiert derzeit bzgl. des funktionalen und des datenorientierten Aspekts das Konzept der *Dekomposition* (siehe Abbildung 4-3). Prinzipiell ist dies auch für den organisatorischen wie operationalen Aspekt möglich. Im Bezug auf die Erstellung von *Teilmodellen* ist dies bei AOPM unter funktionaler Betrachtung möglich. Der oberste komposite Prozess in der Hierarchie stellt das Gesamtmodell dar, die erste Ebene darunter bildet die Teilmodelle mit ihrem obersten kompositen Prozess ab.

Die hier beschriebenen Lösungsansätze dienen vor allem dazu, die große Menge an Modellierungselementen in den jeweiligen Prozessmodellen zu strukturieren. AOPM, BPMN und EPK liefern dazu entsprechende Ansätze.

Die Darstellung der unterschiedlichen Out- und Input-Beziehungen erfolgt über die explizite Abbildung der Datenflüsse. Zwar werden damit alle nötigen Informationen geliefert, eine übersichtliche Darstellung ist damit jedoch nicht möglich.

Für die notwendige Erfassung der Abhängigkeiten zwischen oder innerhalb der einzelnen Aspekte ist es möglich, dies über Constraints zu beschreiben, die dann zur Ausführungszeit aufgelöst werden. Um weiterhin primär im Rahmen des Designs und der Modellierung zu bleiben, sei hier zum Beispiel auf [IJMZ10] und die darin vorhandenen Referenzen auf weitere Ansätze verwiesen.

4.2.4 Zeitlicher Aspekt

Werden Prozesse modelliert, geht es primär um die einzelnen Schritte sowie deren Beziehung zueinander. Darüber hinaus sind der datenorientierte, der organisatorische oder der operationale Aspekt von Interesse [JaBu96] [CuKO92]. Für ein besseres Verständnis und vor allem eine bessere Planung der späteren Ausführung (vor allem im Hinblick auf das Projektmanagement [Burg02]) ist jedoch auch die Angabe eines zeitlichen Aspektes sinnvoll. Beispiele sind in Tabelle 4-4 zu sehen.

	Beschreibung
Beispiel (1)	Die administrative Aufnahme eines Patienten in einer Klinik sollte nicht mehr als 10 Minuten in Anspruch nehmen.
Beispiel (2)	Der Zeitraum zur Einrichtung eines Studiengangs beträgt circa 12 Monate.
Beispiel (3)	Die Vorbereitungen zum Senatsbeschluss in der Universität Bayreuth sollen eine Woche vorher beginnen.

Tabelle 4-4 Beispiele zeitliche Perspektive

Anforderungen

Prozessschritte werden üblicherweise mit fixen geometrischen Abmessungen dargestellt. Diese Angaben stellen allerdings rein inhaltliche Informationen dar, die nichts darüber aussagen, wie lange ein einzelner Schritt dauert. Selbiges gilt für Prozessabschnitte. Je mehr Prozesse sequentiell hintereinander geschaltet werden, desto länger wird die Prozesskette bzgl. der graphischen Darstellung (nicht zwingend auch die dazugehörige Ausführungszeit). Eine *Zeitangabe an den einzelnen Schritten*, elementar wie komposit, eines Prozessmodells sollte daher möglich sein.

Die Angabe eines *Zeitfaktors ist auch im Bezug auf einen Prozessabschnitt* sinnvoll. Statt nur auf der darüber liegenden kompositen Ebene eine Zeitangabe direkt am kompositen Prozess zu definieren, ist es sinnvoll, dies auch auf der darunter liegender Ebene zu ermöglichen. Dann muss jedoch auf die Konsistenz der Zeitangaben geachtet werden. Dem Anwender wird eine bessere Übersicht über den Gesamtprozess geboten, wenn an den einzelnen Schritten der darunter liegenden Ebene(n) zusätzliche Zeitangaben gemacht werden. Die Angabe des zeitlichen Aspektes kann *textuell* oder *graphisch* erfolgen.

Es sollte möglich sein, konkrete *Start- und/oder Endzeiten* für einen Prozess zu definieren.

Lösungsansätze

Eine mögliche Lösung ist die Angabe von Zeiten als *Attribut* eines Prozesselements. Für jeden Schritt ist damit eine individuelle Zeitangabe in textueller Form möglich (graphisch wäre nur die Anzeige des Aspektes an sich leicht nachvollziehbar darstellbar). Die Gesamtzeit muss sich der Anwender aus der Summe der einzelnen Zeitangaben ableiten, es sei denn sie wird im übergeordneten Prozess angegeben. Je nach Darstellungsform sind die Attribute nicht auf den ersten Blick im Prozessmodell zu sehen, sondern in einer Zusatzansicht verborgen. Andernfalls stellen sie ein weiteres, platzeinnehmendes Informationselement auf dem Prozessblatt dar.

Es können auch separate *Kommentar- oder Anmerkungsfelder* verwendet werden, die nicht per se einem Prozessschritt zuzuordnen sind. Hierbei wird die textuelle und graphische Angabe kombiniert. Der Zeitaspekt kann sowohl für einzelne Schritte als auch für einen Prozessabschnitt spezifiziert werden. Da solche Felder jedoch meist für die Angaben von Informationen jeglicher Art, die nicht mit einem Modellierungskonstrukt dargestellt werden können, verwendet werden können, handelt sich um keine eindeutige Darstellung (siehe dazu [RIRG09]).

Eine weitere graphische Möglichkeit ist der *Zeitstrahl*, entlang dessen die Prozessschritte positioniert werden. In Kontrollflussrichtung verlaufend, optional mit Zeitabschnitten markiert, entspricht die Länge des Zeitstrahls dem Zeitraum, der für die Durchführung der angezeigten Prozessschritte benötigt wird. Die Positionierung der Schritte orientiert sich an den eingetragenen Zeitabschnitten. Individuelle Zeitangaben wären über ein Attribut zusätzlich möglich (siehe oben).

Bei den **EPKs** sind weder graphisch noch textuell explizite Zeitangaben möglich. Sie sind somit für die Modellierung zeitkritischer Prozessmodelle nicht gut geeignet.

Auch bei den Aktivitätsdiagrammen der **UML** sind in keiner Weise Modellierungselement oder Eigenschaften definiert, die zur Abbildung eines zeitlichen Aspektes verwendet werden können.

Bei **BPMN** sind Zeitangaben über die Modellierung von sogenannten Timer-Events möglich, die textuell durch konkrete Zeitangaben ergänzt werden können. Ein Zeitstrahl als Modellierungselement ist nicht definiert, jedoch Kommentare bzw. Textfelder, die zur Angabe von Zeiten verwendet werden können.

AOPM bietet die Möglichkeit, über Attribute oder Kommentarfelder Zeitangaben zu spezifizieren. Ein Zeitstrahl ist in Form der Phasen definiert. Jede Phase kann mit einer individuellen Länge dargestellt werden. Dabei gilt es das entsprechende Verhältnis zwischen der Breite des Prozessblattes und der Länge der Phase zu definieren. Die (Sub-) Prozesse sind jedoch hierarchisch eine Ebene tiefer und somit nicht auf demselben Prozessblatt modelliert.

Von keiner der vorhandenen Prozessmodellierungssprachen werden die Anforderungen bezüglich eines zeitlichen Aspektes somit (vollständig) umgesetzt. Entweder fehlt die Möglichkeit zur Zeitangabe komplett (wie zum Beispiel bei den EPKs), oder es fehlt die Möglichkeit zur Darstellung des zeitlichen Aspektes bezogen auf den gesamten Prozessablauf.

4.2.5 Komplexe Entscheidungssituation

Nicht nur bei fachspezifischen Prozessen, so zum Beispiel im naturwissenschaftlichen Bereich oder in der Medizin, treten oftmals diffizile Entscheidungssituationen auf (siehe dazu

Tabelle 4-5). Entscheidungen werden unter Abwägung diverser Informationen und Kriterien getroffen. Die Herausforderung dabei ist, die Übersicht zu behalten.

	Beschreibung
Beispiel (1)	Medizinische Entscheidungen beruhen auf mehreren Kriterien: Ob eine Behandlung bzw. Therapie durchzuführen ist oder nicht, ob zur Klärung eine erweiterte Diagnose notwendig ist oder ob ein Behandlungsziel erreicht werden konnte oder nicht (vgl. [Meil05]).
Beispiel (2)	Die Entscheidung über die Einführung eines Studiengangs auf Universitätsebene beinhaltet Beschlüsse des Fakultätsrates, des Senates sowie der LuSt (= Präsidialkommission für Lehre und Studium) [UnBa11], die jeweils eine Vielzahl von Unterlagen zu bewerten haben.

Tabelle 4-5 Beispiele komplexer Entscheidungssituationen

Anforderungen

Eine *Vielzahl von elementaren Einzelentscheidungen* ist zur Erfassung relevanter Kriterien notwendig und muss bei der Abbildung in einem Prozessmodell *prägnant* abgebildet werden.

Die Einzelentscheidungen sind *nicht unabhängig voneinander* zu sehen, sondern führen in ihrer Gesamtheit bzw. Kombination zu einer Gesamtentscheidung.

Das Ergebnis der Gesamtentscheidung bestimmt den weiteren Verlauf des Prozesses, wobei *mindestens zwei verschiedenen Ausgänge* möglich sein müssen.

Die *Angabe entscheidungsrelevanter Informationen* muss gegeben sein, ebenso die Angabe *von Ergebnissen* der Entscheidung(en), wenn diese dokumentiert werden sollen.

Die Angabe von Rollen zur Spezifikation der für die Entscheidung(en) Verantwortlichen muss möglich sein.

Lösungsansätze

Jede Einzelentscheidung kann als separater *Entscheider oder Konnektor* modelliert und mit den anderen Entscheidungen logisch verknüpft werden. Die möglichen Verläufe im Anschluss an eine Entscheidung werden dargestellt, indem die Entscheider mit Kontroll- oder Datenflüssen verbunden werden, wie es in Abbildung 4-4 schematisch zu sehen ist.

Dieser Lösungsansatz kann mit den in Kapitel 3.3.1 genannten Prozessmodellierungssprachen realisiert werden. So sind entsprechend Konnektoren (**EPK**, **AOPM**) und Verzweigungen (**UML** Aktivitätsdiagramm) bzw. Gateways (**BPMN**) vorgesehen. Eine prägnante Modellierung des Sachverhaltes ist damit aber nicht möglich. Denn es können keine Rollen angegeben werden, um anzuzeigen, wer für eine Entscheidung verantwortlich ist.

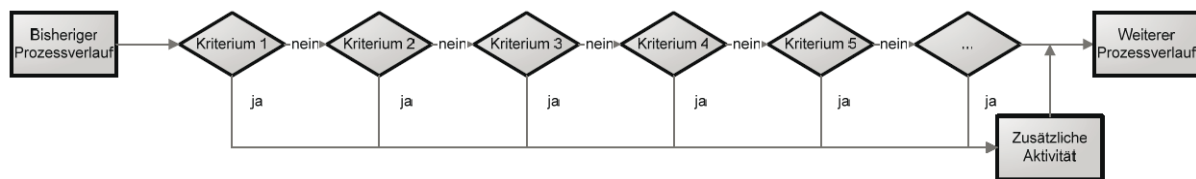


Abbildung 4-4 Komplexe Entscheidung mittels elementaren Entscheidern (AOPM) aus [Meil05]

Speziell für den medizinischen Kontext hat Meiler in [Meil05] für AOPM den sog. *Evidenzbasierten Entscheider* ausgearbeitet. Wie in Abbildung 4-5 zu sehen ist, sind die einzelnen Entscheidungen bzw. ihre Kriterien zwar einzeln aufgelistet, aber insgesamt als eine Entscheidung dargestellt. Aufgrund besonderer Anforderungen aus dem medizinischen Kontext heraus, gibt es genau zwei Ausgänge, je nachdem, ob die Kriterien erfüllt sind oder nicht; weitere wären denkbar und möglich. Das Konstrukt wird über den Datenfluss in den Gesamtprozess mit eingebunden, so dass Input- und Outputdaten angegeben werden können. Rollen können allerdings auch hier nicht angegeben werden. Im Gegensatz zur obigen Darstellung ist jedoch insgesamt eine prägnante Darstellung möglich.

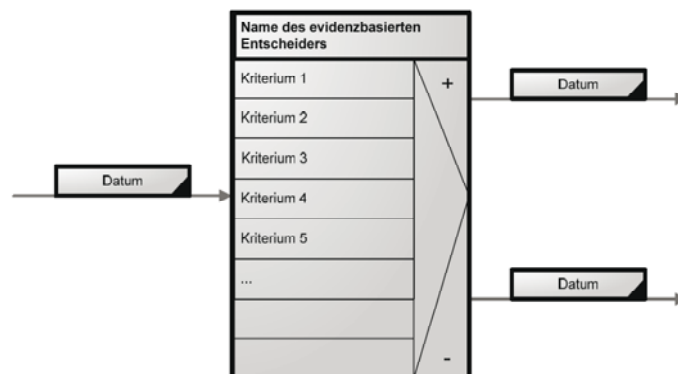


Abbildung 4-5 Evidenzbasierter Entscheider aus [Meil05]

4.2.6 IT-relevante Prozessabschnitte

Prozessmodelle dienen dazu, den Stakeholdern einen Überblick über die zur Erfüllung einer Aufgabe auszuführenden Aktivitäten, die Anwendungsprozesse, zu geben [FrvL03] [Maur96]. Die spätere Prozessausführung erfolgt zum Großteil mittels informationstechnischer Unterstützung (Tabelle 4-6). Die technische Umsetzung des Anwendungsprozesses wird als Workflow bezeichnet [FrvL03] [JaBu96] [Maur96]. Sowohl der Anwendungsprozess als auch der Workflow bilden inhaltlich prinzipiell dasselbe ab.

	Beschreibung
Beispiel (1)	Die Durchführung der Leistungsabrechnung im Krankenhaus erfolgt mit Hilfe des KIS-Systems (Krankenhaus-Informationssystem).
Beispiel (2)	Die Anmeldung zu Prüfungen, Seminaren oder Kursen an der Universität Bayreuth erfolgt über das sog. FlexNow, das elektronische Prüfungssystem.

Tabelle 4-6 Beispiele ausführbare Prozessabschnitte

Anforderungen

Für den Anwender sollen die Darstellungen im Prozessmodell möglichst *intuitiv* und in sich verständlich sein. Eine graphische Modellierungssprache ist dabei zu bevorzugen.

Alle Informationen in einem Prozessmodell, die für die IT-technisch unterstützte Ausführung relevant sind, müssen *explizit formal definiert* sein, syntaktisch wie auch semantisch.

Das Prozessmodell zur Ansicht für den Anwender und das Prozessmodell für die informationstechnische Umsetzung müssen *prinzipiell denselben Inhalt abbilden*.

Lösungsansätze

Grundsätzlich kann unterschieden werden, ob ein Prozessmodell, das zur Ansicht für den Anwender erstellt wurde, sofort IT-technisch umgesetzt werden kann, oder aber erst in eine ausführbare Sprache übertragen werden muss [Böhm00] [RiSt04] [Ste99]. Bei den Modellierungssprachen, die im Rahmen dieser Arbeit betrachtet werden, scheint letzteres die gängigere Methode zu sein, obwohl ersteres weitaus leichter zu realisieren ist. (Für weitere Details sei dazu zum Beispiel auf [Böhm00] und [Ste99] verwiesen.) Hier soll lediglich aufgezeigt werden, dass bereits bei der Auswahl der Modellierungssprache Aspekte der Ausführung zu berücksichtigen sind.

Allgemein ist festzuhalten, dass die Prozessmodelle für die Anwender vorzugsweise in graphischer Form erstellt werden [CuKO92]. Bei den in Kapitel 3.3.1 vorgestellten Modellierungssprachen handelt es sich ohne Ausnahme um solche. Zur Definition ausführbarer Modellierungssprachen sind in den meisten Fällen vor allem zusätzliche implementierungstechnische Detailinformationen nötig [Allw06] [Böhm00] [Ste99]. Diese können sich auf formale Aspekte (zum Beispiel bzgl. der syntaktischen Korrektheit), wie auch auf inhaltliche Aspekte (zum Beispiel bzgl. Detaillierungsgrad und Präzision) beziehen [Böhm00].

Die Aktivitätsdiagramme der **UML** sind zunächst nicht ausführbar. Sie müssen in einem extra Schritt in ein IT-technisch umsetzbares Modell übertragen werden [Schä10].

Auch **EPKs** fokussieren die graphische Notation der Prozesse. EPK-Modelle können zum Zweck der Ausführung nach BPEL (= Business Process Execution Language) [IBM11] abgebildet werden. Wie [StIv07] zu entnehmen ist, sind dabei jedoch einige Einschränkungen im Bezug auf den Informationsgehalt in Kauf zu nehmen.

Bei **BPMN** wird zur ausführbaren Beschreibung der Prozesse WS-BPEL oder XPD, beides XML-basierte Formate, verwendet [FrRH10]. Für BPEL definiert der BPMN-Standard, wie ein BPMN-Diagramm in BPEL übersetzt werden sollte. Das Mapping ist in keiner Weise trivial, zudem ist keine vollständige Übertagung der Informationen aus dem BPMN-Modell möglich; es fehlt vor allem an technischen Detailinformationen (siehe zum Beispiel [OvAD06], [BPMN2.0]). Die Übersetzung nach XPD wurde von der WfMC beschrieben (siehe [WoMC11]).

AOPM wird aktuell nur für die graphische Modellierung von Prozessen verwendet. Im Rahmen des FORFLOW-Projektes wurde zum Beispiel BPEL zur Modellierung ausführbarer Prozesse verwendet (siehe [MePa08], Teilprojekt 4.2.).

4.2.7 Sprünge

Sprünge in einem Prozessmodell stellen Abweichungen des ansonsten kontinuierlich fortlaufenden Kontrollflusses dar. Sie dienen zur Ergebnisverbesserung, indem diese durch erneute Ausführung der dazu notwendigen Schritte überarbeitet werden (= *Rücksprung*); andersherum kann der Prozessverlauf verkürzt werden, in dem Schritte ausgelassen werden (= *Vorsprung*), wie es in Abbildung 4-6 zu sehen ist. Den Ansatzpunkt der Sprünge wird als *Quelle* bezeichnet; der Wiedereinstiegspunkt als *Senke*. Konkrete Beispiele sind in Tabelle 4-7 zu finden.

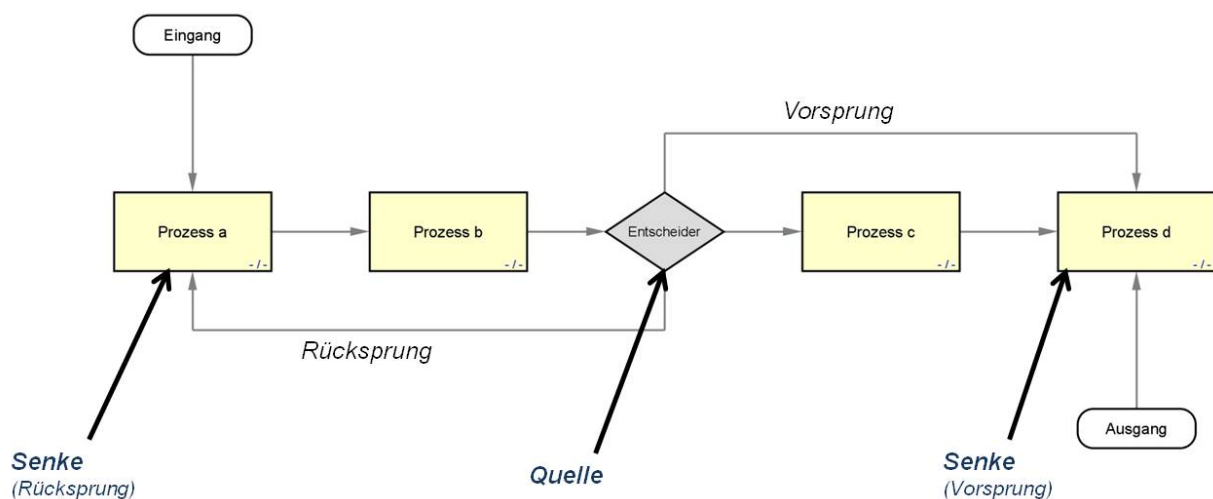


Abbildung 4-6 Sprünge im Prozessmodell

Es handelt sich bei den Sprüngen prinzipiell um Alternativen, die jedoch im Falle eines Rücksprungs im bereits durchgeführten Teil des Prozesses liegen und im Falle eines Vorsprungs unter gewissen Bedingungen Schritte auslassen, die prinzipiell für die Ausführung vorgesehen sind.

	Beschreibung
Beispiel (1)	Die Entscheidung über die Einrichtung des Studiengangs an der Universität Bayreuth ist ein iterativer Prozess. Die notwendigen Unterlagen müssen meist mehrmals überarbeitet werden, bis sie vollständig genehmigt sind.
Beispiel (2)	Ist nach einer Hüft-Operation die Einschaltung des Sozialdienstes/ der Pflegeüberleitung nicht notwendig, kann dieser Schritt übersprungen werden.

Tabelle 4-7 Beispiele Sprünge

Anforderungen

Es muss zwischen Vorsprüngen und Rücksprüngen unterschieden werden.

Die Sprünge beziehen sich sowohl auf den *Kontroll*-, als auch auf *den Datenfluss*. Letzteres weil die Entscheidung für einen Rück- oder Vorsprung auf der Basis von entsprechenden Informationen zu treffen ist. Beide Flussarten müssen also im Kontext der Sprünge definiert werden können.

Als *Quellen* sind Kontroll- bzw. Datenflusselemente wie Konnektoren/ Entscheider möglich, als *Senken* elementare oder komposite Prozessschritte sowie Kontrollflusselemente.

Für Quellen muss es erlaubt sein, mehr *als einen ausgehenden Fluss* zu definieren: einen mit dem der Anwender dem normalen/ Default Kontrollfluss folgt und zum direkt nachfolgenden Prozessschritt übergeleitet wird; einen Weiteren der die Vor- oder Rücksprünge modelliert.

Bei den Senken soll *mehr als ein eingehender Fluss* erlaubt sein, da von mehreren Stellen innerhalb des Prozessablaufes zu ein- und derselben Stelle gesprungen werden kann.

Da sich vor allem bei großen Prozessmodellen mit langen Abläufen, Sprünge nicht unbedingt auf ein Prozessblatt beschränken, wie es in Abbildung 4-6 zu sehen ist, muss eine *prozessblattübergreifenden Darstellung* möglich sein. Dabei ist vor allem die Verdeutlichung der Zusammengehörigkeit der jeweiligen Quellen und Senken wichtig. Bei einem hierarchischen Prozessmodell müssen Start- und Endpunkte der Vor- oder Rücksprüngen *auf jeder Ebene* zu modellieren sein.

Die Verbindung zwischen Quelle und Senke muss *eindeutig erkennbar und nachvollziehbar* sein.

Lösungsansätze

Solange sich die Sprünge auf ein Prozessblatt beziehen, wie es auch in Abbildung 4-6 zu sehen ist, stellt eine konsistente und eindeutige Modellierung bei keiner Modellierungssprache ein Problem dar. Von einem Entscheider/ Konnektor ausgehend, kann ein Kontrollfluss zur Senke modelliert werden. Das einzige Problem dabei ist, dass so nur der Kontrollfluss dargestellt wird. Bei prozessblattübergreifenden Sprüngen verhält sich die Situation anders.

In einem Aktivitätsdiagramm der **UML** können einer Aktivität beliebig viele *Startknoten* zugeordnet werden. Die Diagramme können zudem mehrere *Endknoten* besitzen. Mit einem solchen Endknoten wird jedoch das vollständige Ausführungsende eines Prozess angezeigt; danach dürfen keine weiteren Aktivitäten mehr folgen. Damit ist eine Modellierung von Sprüngen nicht möglich. Weitere Konstrukte, die zur Darstellung der Sprünge verwendet werden könnten, sind nicht definiert.

Bei den **EPKs** gibt es sog. *Prozesswegweiser*, die anstelle einer Funktion modelliert sind als Hinweis auf andere Prozesse. Durch die Angabe einer Referenz setzen sie mindestens zwei verschiedene Prozesse miteinander in Beziehung. Somit könnten sie zur Darstellung von prozessblattübergreifenden Sprüngen, explizit zur Darstellung der Quellen, verwendet werden. Die Quellen können an jeder beliebigen Stelle im Modell platziert werden, die Angabe der Senken ist jedoch auf ganze Prozesse und damit die dazugehörigen Prozessanfänge begrenzt. Sprünge in die Mitte eines Ablaufs hinein sind somit nicht möglich. Abhilfe kann dabei eine Restrukturierung des Prozessmodells schaffen, so dass die Senke als erster Schritt eines Teilprozessmodells definiert ist.

Die Standardmodellierungssprache **BPMN** definiert diverse sogenannte Start, Intermediate und End Events (siehe auch Anhang B), jeweils mit eigener graphischer Darstellungsform. Sie zeigen ein Ereignistyp am Anfang, innerhalb bzw. Ende eines Prozesses an.

Bei wenigen Events besteht jedoch die Möglichkeit zur Definition einer eindeutigen Verbindung von Quelle (= *End Event*) und Senke (= *Start Event*), da es sich um isolierte Einzel-Events handelt, wie zum Beispiel *Timer* oder *Error*. *None* als Start sowie End Event ist sehr allgemein definiert, so dass es zur Darstellung einer Quelle oder Senke verwendet werden kann. Jedoch fehlt es an der Eindeutigkeit des Konstruktes bzgl. des Einsatzzwecks. Mit *Multiple* besteht zwar die Möglichkeit, zu einem End Event mehrere Start Events, die an beliebiger Stelle im Gesamtmodell definiert sind, zu definieren; diese treten dann jedoch auch alle ein, was im Hinblick auf den default Prozessablauf und die dazu modellierten Sprünge nicht erwünscht ist. Bei *Message* und *Signal*, von denen es auch sowohl Start als auch End Event gibt, kann eine Verlinkung definiert werden. Bei *Signal* ist dies nicht zielgerichtet zu einem bestimmten anderen aufnehmenden Signal möglich. Mit *Message* ist dies zwar möglich; doch wird eine Message immer erst über den Umweg eines sog. Participant

gesendet, der diese dann zu einem Prozessschritt weiterleitet. Dies verkompliziert die Darstellung und das Verständnis.

Die *Intermediate Events* sind zur Darstellung von Sprüngen schon per Definition nicht geeignet. Sie beeinflussen den Kontrollfluss nur insofern, als ein für andere Schritte relevantes Ereignis aus dem Prozess herausgelöst wird und der Prozess danach dem normalen Kontrollfluss folgt ohne einen Schritt zu überspringen; schon allein durch diese Vorwärtsausrichtung sind Rücksprünge definitiv von der Darstellung ausgeschlossen.

Als ein Ansatz in die gewünschte Richtung können die sogenannten Off-Page Konnektoren *Link* betrachtet werden, die der Verbindung von Prozessabschnitten (auch auf verschiedenen Prozessblättern, jedoch auf einer Ebene) dienen. Werden mehrere Links verwendet, ist jedoch nicht genau klar, welche Links zusammengehören.

Es ist generell keine Datenflussangabe bei den Events möglich. Die ist ein weiterer Punkt, in dem die genannten Events die Anforderungen nicht erfüllen können. Zudem gilt es zunächst die hohe Anzahl Events und ihre Bedeutung zu verstehen.

Bei Verwendung der Modellierungssprache **AOPM** bieten sich die *externen Ein- und Ausgänge* zur Modellierung der Vor- und/ oder Rücksprünge an. Dabei handelt es sich um Schnittstellen, die sich sowohl semantisch als auch visuell explizit von den normalen Ein- und Ausgängen eines Prozesses abgrenzen. Dabei muss einem externen Ausgang nicht zwingend ein externer Eingang (bzw. umgekehrt) zugeordnet werden, die Kopplung mit einem normalen Ein- oder Ausgang ist ebenfalls möglich. Eine individuelle Benennung dieser Schnittstellen ist möglich, so dass dort Informationen zu der dazugehörigen Quelle oder Senke angegeben werden können. Die Modellierung von Datenflüssen ist zudem möglich.

Es kann somit festgehalten werden, dass AOPM die Anforderungen am besten erfüllt. Bei einer statischen Modellierung sind die Sprünge, wenn sie prozessblattübergreifend abgebildet werden, dennoch auch bei AOPM schwer nachvollziehbar. Eine entsprechende Funktionalität des Modellierungswerkzeugs kann dort weitere Unterstützung bieten.

4.2.8 Unterschiedliche Sichtweisen

In den meisten Fällen sind an der Durchführung eines Prozesses unterschiedliche Rollen beteiligt (siehe Beispiele in Tabelle 4-8). Je nach Prozess kann damit ein häufiger Rollen- bzw. Sichtwechsel verbunden sein, was vor allem bei stark funktionsorientiert ausgerichteten Organisationsstrukturen der Fall ist. Die verschiedenen Rollen sind dabei oftmals mehrfach im Laufe des Prozesses an der Ausführung beteiligt.

Beschreibung	
Bei der Durchführung eines Studiengangs erfolgt ein häufiger Wechsel zwischen den Studierenden sowie den verschiedenen Abteilungen und	Beispiel (1)

	Verantwortlichen der Universität (Prüfungsausschuss, Prüfungsamt, Referate, Dezernate, Studiengangsmoderator, ...).
Beispiel (2)	Bei der Behandlung eines Patienten in einer Klinik erfolgt ein Wechsel zwischen dem Patienten und den pflegenden bzw. behandelnden Personen (Arzt, Pfleger, Physiotherapeut, Medizinisch-technischer Assistent, ...).

Tabelle 4-8 Beispiele unterschiedlicher Sichtweisen

Anforderungen

Es muss klar sein, *welche Schritte von welcher Rolle* ausgeführt werden sollen.

Die Rollenwechsel müssen für den Anwender beim Lesen des Prozessmodells *ersichtlich* sein. Er muss dabei erkennen können welche Schritte für ihn selbst *relevant* sind und welche von anderen Verantwortlichen ausgeführt werden.

Der Rollenwechsel kann nach beliebig vielen Schritten bzw. nach jedem Schritt vorgenommen werden.

Die *Darstellung des eigentlichen Prozessablaufs (mittels Daten- bzw. Kontrollfluss)* darf *nicht vernachlässigt* werden, da dies für das Gesamtverständnis bzgl. des Prozesses essentiell ist.

Lösungsansätze

Für eine klare Rollenzuordnung ist die *direkte Angabe am eigentlichen Prozessschritt* sinnvoll. Da der Fokus damit auf der Darstellung des Prozessablaufes und den Prozessen an sich liegt, ist ein Rollenwechsel nicht auf den ersten Blick ersichtlich.

Diese Art der Modellierung wird jedoch lediglich von den **EPKs** und **AOPM** unterstützt, nicht jedoch von den **UML**-Aktivitätsdiagrammen oder **BPMN**.

Ein weiterer Ansatz ist, den Fokus der Darstellung hin zum organisatorischen Aspekt zu verschieben. Dieser Gedanke wird in Form der sog. *Swimlanes* realisiert. Eine Swimlane stellt einen horizontalen oder vertikalen Balken dar, dem jeweils eine Rolle zugewiesen wird. Entlang dieser Swimlanes werden die Prozesse entsprechend ihrer Rolle angeordnet, wie es in Abbildung 4-7 zu sehen ist. Daten und Datenflüsse sind hier der Übersichtlichkeit wegen nicht modelliert. Mit der Swimlane-Darstellung sind die Rollenwechsel gut zu erkennen, doch ist der Prozessablauf weniger intuitiv zu erfassen, vor allem wenn mehrere Verzweigungen vorkommen.

Bei den Aktivitätsdiagrammen der **UML** können Swimlanes modelliert werden, um den einzelnen Aktivitäten Rollen zu zuordnen.

Bei den **EPKs** steht primär die Darstellung einer Prozesskette im Vordergrund, weniger die Darstellung des organisatorischen Aspektes. Somit werden die Rollen direkt an den einzelnen Funktionen angegeben; die Darstellung von Swimlanes ist in Form von sog. *Funktionsbändern* möglich.

Obwohl auch **BPMN** die Darstellung der zeitlogischen Abfolge von Aktivitäten fokussiert, sind hier, im Gegensatz zu den EPKs, mit den Pools und Lanes gleich zwei Typen von Swimlanes definiert. Damit sind Rollenwechsel in einem BPMN-Prozessmodell stets gut zu erkennen.

Für **AOPM** ist eine Version vorhanden, in der neben der direkten Rollenangabe an einem Prozessschritt, Swimlanes modelliert werden können. Der Anwender muss sich dann aber vor der Modellerstellung entscheiden, ob er mit Verwendung der Swimlanes den Schwerpunkt auf den organisatorischen Aspekt legen möchte oder die zeitlich-logische Abfolge der Funktionen im Fokus sieht.

Beide Darstellungsformen haben somit ihre Vor- und Nachteile. Bei reiner Betrachtung der Modellierungssprachen, muss bei der Auswahl der Modellierungssprache abgewägt werden, welche Darstellung zu bevorzugen ist; Bei AOPM sind, unter Verwendung der neusten Version des i>pm (siehe dazu [Volz11]), beide Darstellungsformen für die Anzeige ein- und desselben Inhalts möglich.

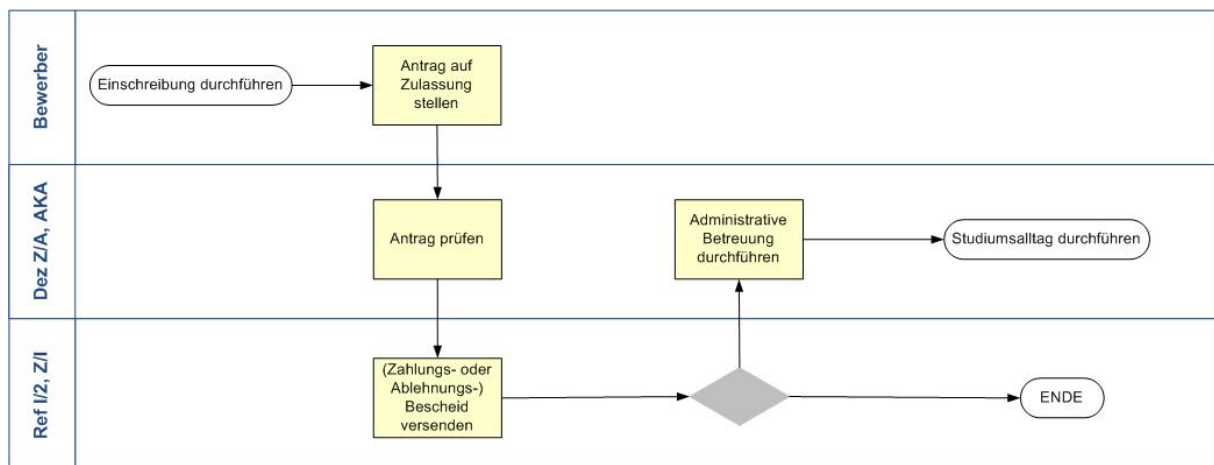


Abbildung 4-7 Darstellung eines Prozesses mit Swimlanes

4.2.9 Unterschiedliche Datentypen

Im Rahmen eines Prozesses fallen für die verschiedenen Schritte viele Input- und Outputdaten an. Dabei kann es sich um elektronische oder Papier-Dokumente handeln, um physische Gegenstände wie zum Beispiel Chipkarte oder auch um mündliche Informationen in Form von Mitteilungen. Werden alle Daten auf dieselbe Art und Weise dargestellt, kann das Lesen

der Modelle bzgl. des datenorientierten Aspektes sehr mühsam werden; die Unterschiede müssen implizit vom Leser erfasst werden.

	Beschreibung
Beispiel (1)	In einem klinischen Behandlungsprozess fallen unterschiedliche Arten von Daten an: die Krankenkassenkarten der Patienten bei der Aufnahme, elektronische Dokumente im KIS, Röntgenbilder oder EKG-Aufzeichnungen.
Beispiel (2)	Bei der Durchführung des Studiengangs fallen unterschiedliche Arten von Daten an. So werden die Daten der Studenten elektronisch erfasst, dazu gibt es aber auch noch den physisch fassbaren Studentenausweis. Zudem muss der Student die erforderlichen Studienbeiträge an die Universität überweisen; die dabei anfallenden Daten gilt es ebenfalls zu erfassen.

Tabelle 4-9 Beispiele Datentypen

Anforderungen

Eine *prägnante Unterscheidung* der verschiedenen Datentypen ist notwendig.

Die Darstellung der Unterschiede soll *visuell* erfolgen und direkt im Modell zu erkennen sein.

Es muss zu erkennen sein, dass die verschiedenen Datentypen *zur Gruppe der Daten gehören*.

Lösungsansätze

Es besteht die Möglichkeit an der Modellierungssprache direkt anzusetzen, in der nicht nur ein Datenelementtyp definiert wird, sondern verschiedene.

Bei den Aktivitätsdiagrammen der **UML** werden jedoch lediglich Objektknoten zur Darstellung von Daten oder Informationen definiert.

Bei den **EPKs** gibt es Informationsobjekte mit unterschiedlicher Bedeutung und jeweils eigener Darstellungsart (siehe Tabelle 3-1 in 3.1.1). Es werden elektronische Dokumente, Dokumente in Papierform und Informationsobjekte im Allgemeinen unterschieden. Damit sind bereits wesentliche Unterscheidungen getroffen, auch wenn die Unterteilung nicht individuell nach den eigenen Bedürfnissen vorgenommen werden kann. Da alle Elemente an die Form eines Rechteckes angelehnt sind, ist eine Zusammengehörigkeit erkennbar.

BPMN definiert lediglich sog. Data Objects. Damit werden sowohl elektronische Objekte wie Dokumente oder Datensätze, als auch die physische Objekte aus der realen Welt wie die Krankenkassenkarte eines Patienten oder das Produkt eines Fertigungsprozesses dargestellt.

Ebenso definiert auch **AOPM** lediglich eine Art von Datum. Zusätzlich gibt es lediglich das Konstrukt einer Checkliste; doch handelt es sich hier um ein sehr spezielles Konstrukt, das

nicht unbedingt allgemeingültig anwendbar ist. Es besteht jedoch die Möglichkeit, die Daten zu kategorisieren. Die Modellierungselemente der Daten können *der Kategorie entsprechend eingefärbt werden*, sowie es auch in den Modellierungsprojekten des Lehrstuhls erfolgt ist, die mit AOPM modelliert worden sind (siehe zum Beispiel Abbildung 2-2 und Abbildung 2-3).

4.2.10 Unterschiedliche Prozesstypen

Innerhalb einer großen Organisation fällt oftmals eine Vielzahl unterschiedlicher Prozesse an, im medizinischen Kontext zum Beispiel neben den medizinischen Prozessen auch verwaltungstechnische Prozesse. Letztere fallen auch im Rahmen der Produktentwicklung neben den Entwicklungsprozessen an. Die unterschiedlichen Prozesstypen unterscheiden sich in wesentlichen Charakteristiken, wie zum Beispiel manuelle oder automatisierte Ausführung. Wie bei den verschiedenen Datentypen in Kapitel 4.2.9 kann die visuelle Hervorhebung der unterschiedlichen Typen von Interesse sein. Im Bezug auf Beispiel (2) in Tabelle 4-10 ist damit zum Beispiel bei apparativen Untersuchungen zusätzliches Personal für die Durchführung notwendig.

	Beschreibung
Beispiel (1)	Neben den fachspezifischen Prozessen (zum Beispiel medizinisch, entwicklungstechnisch, ...) fallen in großen Organisationen auch meist verwaltungstechnische Prozesse an.
Beispiel (2)	Bei den medizinischen Prozessen einer Klinik kann nach apparativ und nicht-apparativ Untersuchungen unterschieden werden.

Tabelle 4-10 Beispiele Prozesstypen

Anforderungen

Eine *prägnante Unterscheidung* der verschiedenen Prozesstypen ist notwendig.

Die Darstellung der Unterschiede soll *visuell* erfolgen und direkt im Modell zu erkennen sein.

Die verschiedenen Prozesstypen müssen dennoch *als Prozess zu erkennen* sein.

Lösungsansätze

Bei den im Rahmen dieser Arbeit berücksichtigten Modellierungssprachen (siehe Kapitel 3.3.1) wird in den Definitionen der Modellierungssprachen nicht zwischen verschiedenen Prozesstypen unterschieden.

Bei AOPM besteht jedoch die Möglichkeit, Prozesse entsprechend festgelegter *Kategorie einzufärben*, wie es bereits im Zusammenhang mit den verschiedenen Datentypen in Kapitel

4.2.9 erfolgt ist. In dem Projekt zur Modellierung der universitären Prozesse wurde dies bereits im Hinblick auf die Unterscheidung von Hauptprozessen und Arbeitsschritte angewendet, die aber zusätzlich noch auf verschiedenen Ebenen des Prozessmodells modelliert wurden.

4.3 Fazit – Handlungsbedarf

Das Kapitel hat eine Vielzahl von unterschiedlichen Anforderungen aufgelistet, die es in der Designphase des Prozesslebenszyklus, also vor der Modellierung, zu identifizieren und anschließend umzusetzen gilt. Es handelt sich um Beispiele aus den Projekten des Lehrstuhls für angewandte Informatik IV an der Universität Bayreuth, weshalb eine Generalisierung nicht in allen Fällen möglich ist. Es besteht zudem nicht der Anspruch, für alle diese identifizierten Anforderungen eine Lösung zu entwickeln.

Vielmehr sollte aufgezeigt werden, wie vielfältig die Anforderungen und ihre Umsetzungsmöglichkeiten sind. Es wird nachvollziehbar, dass die Umsetzung nicht mittels einer einzigen universellen Modellierungssprache und zudem nicht immer mit den bereits vorhandenen Modellierungssprachen und ihren bereits festgelegten Eigenschaften möglich ist. In manchen Fällen kann es auch sinnvoll sein, das Modellierungswerkzeug in seiner Funktionalität zu erweitern. Während diese Aspekte alle eindeutig der *Designphase* bzw. der Teilphase der Gestaltung zuzuordnen sind (siehe Abbildung 1-2), ist zu berücksichtigen, dass sich die finalen Handlungsmöglichkeiten in gewissen Teilen nichtsdestotrotz auch auf den Modellierungsvorgang an sich beziehen können.

Ziel muss es sein, die zunächst ungeordnete Menge an *Anforderungen und ihre Umsetzungsmöglichkeiten* zu strukturieren. Mit der Meta-Modellierungsmethode aus Kapitel 3.3.2 erfolgt eine *Strukturierung von (Meta) Modellen*. Es bietet sich an, diese Strukturierung auch für das Design unter Einbeziehung des Modellierungsvorgangs zu verwenden. Die Meta-Modellierungsmethode aus Kapitel 3.3.2 unterscheidet zwischen der Definition sowie Verwendung einer Modellierungssprache. Dadurch kann garantiert werden, dass sich die Umsetzung der Anforderungen nicht (nur) auf die Modellierung bezieht, sondern gezielt auf die Charakteristiken einer Modellierungssprache eingegangen wird und auch Aspekte der Ausführung beachtet werden. Das entwickelte *Rahmenwerk und Vorgehensmodell für das Design* wird im nun folgenden Kapitel 5 vorgestellt.

Kapitel 5

Design und Modellierung – im Rahmen der Meta Modell Hierarchie

Ziel der Arbeit ist es, das Design mehr in den Fokus der Betrachtung zu stellen. Dazu wurde ein methodisches Rahmenwerk inklusive einem Vorgehensmodell für diese Phase entwickelt. Dies erfolgte in enger Anlehnung an die sich anschließende Phase der Modellierung. Grundlage ist die in Kapitel 3.3.2 vorgestellten Meta-Modell-Hierarchie nach [AkKü01], [AkKü05] und [HeGo06].

In diesem Kapitel erfolgt zunächst die Einordnung der Phasen Design und Modellierung in die Meta-Modell-Hierarchie nach dem derzeitigen „Stand der Technik“, das heißt im traditionellen Sinne. Darauf aufbauend wird eine neuartige Interpretation und Gestaltung von Design und Modellierung entwickelt, die auch explizit auf den Aspekt der Anforderungsanalyse im Rahmen der Designphase eingeht.

5.1 Traditionelle Betrachtung von Design und Modellierung

Nach der traditionellen Interpretation von Design und Modellierung, wie sie in Kapitel 2.3 erläutert wurde, ergibt sich die in Abbildung 5-1 dargestellte Zuordnung der Phasen Design und Modellierung in Bezug auf die Meta-Modell-Hierarchie. Der Vollständigkeit halber werden hier auch die Phasen der Ausführung sowie des Monitorings und Controllings auf der Ebene M0 berücksichtigt.

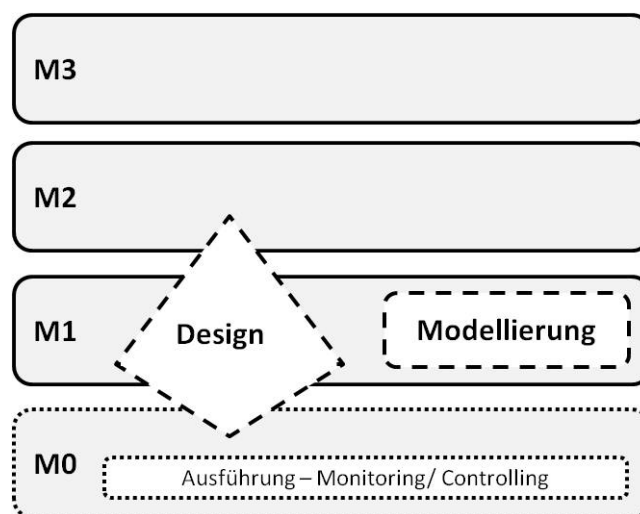


Abbildung 5-1 Traditionelle Einordnung von Design und Modellierung

Modellierung

Die Modellierung bezieht sich gemäß dem traditionellen Prozesslebenszyklus ausschließlich auf Ebene M1 der Meta-Modell-Hierarchie. Dies bedeutet, dass hier unter Verwendung einer Prozessmodellierungssprache die Prozessmodelle erstellt werden, wie sie beispielhaft in Kapitel 2.2 aufgezeigt wurden. Die Modellierung schließt sich der Designphase an, welche zum überwiegenden Teil ebenfalls der Ebene M1 zugeordnet wird.

Design

Das Design wird als initiale Phase des Prozesslebenszyklus betrachtet, vor allem in Bezug auf die Modellierung, die dies bzgl. der Ebene M1 der Meta-Model-Hierarchie zugeordnet werden kann. Auf Basis einer eingehenden Anforderungsanalyse geht es dabei vor allem um die Auswahl einer für den zu modellierenden Sachverhalt adäquaten Modellierungssprache. Meistens werden anwendungsspezifische, das heißt individuell konzipierte Modellierungssprachen bei der Auswahl vernachlässigt und stattdessen (de facto) Standardmodellierungssprachen ausgewählt (siehe dazu zum Beispiel [DGRI06]). Um eine konsistente Modellierung zu garantieren, werden Richtlinien für eine ordnungsgemäße Modellierung vorgegeben [BeRS95], durch die beispielsweise die Anzahl der Hierarchieebenen eines Prozessmodells beschränkt werden.

Auf Ebene M2, der Ebene der Modellierungssprache, bezieht sich das Design somit nur minimal, weshalb in Abbildung 5-1 auch nur die Spitze der Raute diese Ebene berührt. Nur in den seltensten Fällen erfolgt die Definition oder auch nur Anpassung eines Modellierungselements. Dies liegt auch daran, dass bei standardisierten Modellierungssprachen Änderungen nur von den dafür verantwortlichen Organisationen vorgenommen werden dürfen; bei der Standardmodellierungssprache BPMN darf dies zum Beispiel nur die Object Management Group vornehmen.

Ebene M0 wird im Bezug auf das Design ebenso nur in geringem Ausmaß berücksichtigt. Es beschränkt sich auf die Fragestellung, wie die Prozessmodelle im konkreten Anwendungsfall verwendet werden sollen. So kann zum Beispiel festgelegt werden, dass die Durchführung eines Ablageprozesses in der Verwaltung nicht länger als fünf Minuten dauern darf. Im Rahmen des medizinischen Projektes des Lehrstuhls wurde festgelegt, dass die modellierten Prozesse nicht als strenge Handlungsanweisungen bzgl. der Durchführung angesehen werden, sondern mehr der allgemeinen Informationsvermittlung, unter anderem zur Einführung neuer Mitarbeiter dienen.

Ebene M3 ist üblicherweise nicht mit in die Designphase einer Prozessmodellierungssprache einbezogen. Dies ist jedoch als eher unkritisch anzusehen bzw. von geringer Relevanz, da es nicht darum geht den Anwender zu ermutigen, existierende Grundpfeiler der Sprachen zu modifizieren. So basieren die UML (Aktivitätsdiagramme), die EPKs, BPMN sowie AOPM

alle auf dem grundlegenden Paradigma der Knoten und Kanten, obwohl sie unabhängig voneinander entwickelt worden sind.

Bewertung

Es kann festgehalten werden, dass bei der traditionellen Betrachtung der Gestaltungsspielraum im Rahmen des Designs sehr gering ist. In Abbildung 5-1 ist das quantitative Ausmaß des Gestaltungsspielraums graphisch durch die weiße Raute dargestellt. Das aktuelle Vorgehen, bei dem der Fokus auf der Modellerstellung mit einer ausgewählten (das heißt nicht individuell gestalteten) Modellierungssprache liegt, ist als nicht ausreichend anzusehen. Aufgrund der Vielfältigkeit von Anwendungsfällen (bzw. Anwendungssystemen) ist davon auszugehen, dass damit in den meisten Fällen keine erfolgreiche, dem Qualitätsanspruch entsprechende Modellierung gewährleistet werden kann. So konnten [RIRG09] in ihrer Studie feststellen, dass diverse Modellierungssprachen sowohl ein Defizit an Konstrukten als auch nicht eindeutig definierte Modellierungselemente aufweisen. Unter den untersuchten Modellierungssprachen waren unter anderem BPMN und EPKs. Mit einer nur mangelhaften Modellierungssprache kann nicht garantiert werden, dass das Modell die gewünschten Funktionen bzw. Anforderungen erfüllt. Die im nächsten Abschnitt vorgestellte Erweiterung soll dieser Tatsache entgegenwirken.

5.2 Erweiterte Interpretation von Design und Modellierung

Abbildung 5-2 zeigt die aus der neuen Interpretation resultierende Einordnung von Design und Modellierung in die Meta-Modell-Hierarchie.

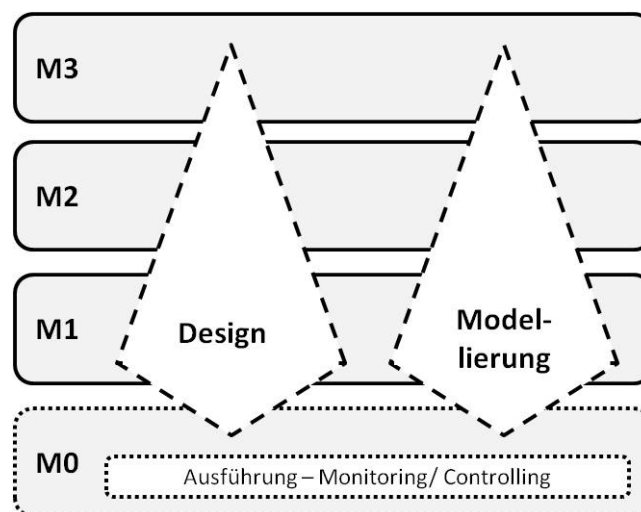


Abbildung 5-2 Erweiterte Interpretation von Design und Modellierung

Modellierung

Modellierung ist der Schwerpunkt der dem Ansatz zugrunde gelegten Meta-Modell-Hierarchie. Das Ergebnis auf jeder Hierarchieebene ist somit ein Modell mit jeweils ebenenspezifischen Eigenschaften:

- M3: abstrakte Modellierungsprimitive
- M2: konkrete Prozessmodellierungssprache(n)
- M1: konkrete Prozessmodelle
- (M0: ausführbare/ ausgeführte Modelle)

Im Unterschied zur traditionellen Vorgehensweise sollen damit in Bezug auf die Modellierung alle Ebenen bzw. Modelle in den neuen Ansatz mit einbezogen werden. Mit dieser Modell-Hierarchie kann dann explizit zwischen der Definition einer Prozessmodellierungssprache auf M3 und M2 sowie der Definition von Prozessmodellen auf M1 unter Verwendung der vorher definierten Prozessmodellierungssprache unterschieden werden. Zudem steht mit den Ebenen M3 bis M1 eine logische Strukturierung der Modelle zur Verfügung. M0 als Teil der Ausführungsumgebung wird der Vollständigkeit halber mit berücksichtigt. Es geht hier um Aspekte bzw. Informationen bzgl. der Modellerstellung, die erst zum Zeitpunkt der Ausführung bekannt sind. Diese sollten sich auf ein Minimum beschränken, da zu diesem Zeitpunkt Änderungen am Modell sehr aufwendig sind.

Die Form und flächige Ausdehnung der Raute in Abbildung 5-2 soll das quantitative Ausmaß der Modellierung verteilt über die einzelnen Ebenen verdeutlichen. Der Schwerpunkt der Modellierung liegt somit auch weiterhin auf M1. Handlungen auf den darüber (und darunter) liegenden Ebenen sind seltener, obwohl doch gerade Ebene M2 mit der Entwicklung von Modellierungssprachen von entscheidender Bedeutung ist. Schließlich wird in dem auf dieser Ebene relevanten Modell die Definition einer Modellierungssprache festgehalten.

Design

Gemäß den Ausführungen zur Modellierung wird nun jede Modellebene, wie es in Abbildung 5-2 zu sehen ist, explizit um den Aspekt des Designs ergänzt. Wie es die graphische Darstellung mit der jeweils nach oben und unten eng zulaufenden Spitze andeutet, ist Bedeutung bzw. das Ausmaß der Designphase auf den einzelnen Ebenen verschieden. Das Design leitet sich aus der Bedeutung und dem Ausmaß der Modellierung auf den verschiedenen Ebenen ab. Auf Ebene M3 findet auch das Design in nur geringem Ausmaß statt, da die dort definierten Artefakte als eher generisch zu betrachten sind. M0 wird in geringem Ausmaß mit einbezogen, da es hier wieder vor allem um Aspekte bzw. Informationen bzgl. der Modellerstellung geht, die erst zum Zeitpunkt der Ausführung bekannt sind, wie zum Beispiel die Angabe konkreter Wartezeiten zum Start eines Prozessschrittes. Der Hauptanteil der Design- und Modellierungsaufgaben entfällt somit zwar

wie schon bei Abbildung 5-1 auf die Ebene M1, doch hat die Designphase in Abbildung 5-2 insgesamt einen weitaus größeren Handlungsspielraum.

Bewertung

Mit Hilfe der Meta-Modell-Hierarchie ist eine Differenzierung sowie klare und zielorientierte Strukturierung der Aufgabe dieser beiden Phasen möglich. Je nach vorliegendem Anwendungsfall können die verschiedenen Ebenen in Anspruch genommen werden.

Es werden nun nicht nur wie bisher Aspekte bzgl. M1 mit dem Fokus auf der reinen Modellerstellung an sich behandelt, sondern es werden auch explizit Aspekte bzgl. einer Modellierungssprache und eines Modellierungswerkzeugs auf M2, M3 und der Ausführung auf M0 Raum gegeben. Der Handlungsspielraum ist somit wesentlich größer.

Die Kopplung bzw. Verbindung zwischen den einzelnen Ebenen erfolgt dadurch, dass in der zugrunde gelegten Meta-Modell-Hierarchie der Inhalt einer höheren Ebene stets auf der darunterliegenden Ebene instanziiert wird [AkKü01]. Für das in dieser Arbeit weiter entwickelte Rahmenwerk wird dies im Bezug auf die Phase der Modellierung direkt übernommen. Durch die enge Kopplung zu der vorausgehenden Designphase kann diese Beziehungseigenschaft im übertragenen Sinn auch für das Design angenommen werden.

Wie in Abbildung 5-2 zu sehen ist, können auch noch auf der Ausführungsebene M0 Aufgaben des Designs und der Modellierung ausgeführt werden. Dies wird damit begründet, dass vor der eigentlichen Umsetzung der Prozessmodelle nur in den seltensten Fällen ausreichend Informationen zur Verfügung stehen werden, um die beiden Phasen des Designs und der Modellierung bereits komplett abschließen zu können. Mit Design und Modellierung auf M0 ist eine *finale Feingestaltung* der definierten Prozesse im Hinblick auf die aktuell vorliegenden Bedingungen während der Prozessausführung möglich, wodurch die Prozessmodelle mit dem letzten „Feinschliff“ versehen werden können.

Es lassen sich somit alle Phasen des Prozesslebenszyklus auf die Meta-Modell-Hierarchie abbilden. Design und Modellierung werden übergreifend den Ebenen M3, M2, M1 und M0 zugeordnet, die Ausführung der Prozesse sowie das Monitoring/ Controlling der Ebene M0. Die Phase Strategie ist außerhalb der Meta-Modell-Hierarchie zu sehen. Denn wie in Kapitel 2.3 beschrieben, wird sie nicht bei jedem Zyklus erneut durchlaufen, sondern wird nur zu Beginn des ersten Zyklus oder bei grundsätzlichen und langfristigen Änderungen, die eine Überarbeitung der Strategie notwendig machen, durchgeführt.

5.3 Klassifikation der Anforderungen

Zu Beginn der Designphase wird zunächst eine grundlegende Anforderungsanalyse bzgl. des vorliegenden Anwendungsfalls durchgeführt. Bei einem Blick in die Literatur ist festzustellen, dass bereits zahlreiche (allgemeingültige) Anforderungen bzgl. Modellierungssprachen und der Modellerstellung an sich analysiert wurden. Bzgl.

Modellierungssprachen sind diese oftmals zusammengefasst in Kriterienkatalogen, mit Hilfe derer die Modellierungssprachen grundlegend entwickelt werden können. Beispielhaft zu nennen wäre hier [FrvL03], die generelle Anforderungen sowie zusätzliche einige spezielle Anforderungen im Bezug auf die Darstellung von Geschäftsprozessen nennen oder [FrPr97] und [Fran98], die Anforderungen im Hinblick auf objektorientierte Modellierungssprachen definieren. Anforderungen bzgl. der Erstellung von Modellen sind zum Beispiel in [BeRS95] oder [MeRA10] zu finden.

Es ist nicht das Ziel dieser Arbeit einen neuen Kriterien- bzw. Anforderungskatalog aufzustellen, auch weil dies für jedes Anwendungsgebiet und jeden Anwendungsfall individuell erfolgen sollte. Vielmehr ist ein Rahmenwerk aufzustellen, das für die genaue Spezifikation der Anforderungen eine Unterstützung bietet.

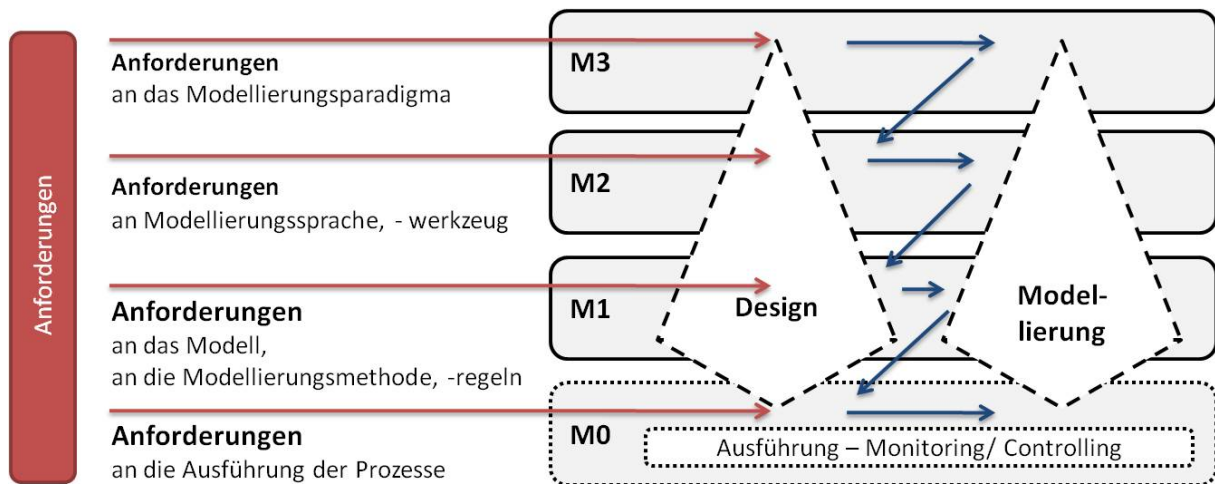


Abbildung 5-3 Klassifikation von Anforderungen gem. der Meta-Modell-Hierarchie

Es gilt die zunächst zu Beginn einer Designphase lose, zum Beispiel im Rahmen eines Brainstormings ermittelten Anforderungen (dies ist in Abbildung 5-3 durch die Box auf der linken Seite dargestellt) im Hinblick auf ihre Umsetzung (die Gestaltungsmöglichkeiten) zielorientiert zu organisieren. Dies erfolgt unter Verwendung der Strukturierung der Design- und Modellierungsphase, wie sie gerade in Kapitel 5.2 beschrieben worden ist. In Abbildung 5-3 ist die Kategorisierung der Anforderungen durch die an der linken Bildseite startenden horizontalen Pfeile dargestellt.

Durch die explizite Strukturierung gemäß den Ebenen der Meta-Modell-Hierarchie kann eine wesentlich präzisere und differenziertere Definition und Abgrenzung von Anforderungen und deren Umsetzung (die Gestaltungsmöglichkeiten) gewonnen werden. Der Anwender wird sensibilisiert über die explizite Definition oder Anpassung einer Prozessmodellierungssprache nachzudenken, so dass er sich nicht nur auf den Modellierungsvorgang auf M1 beschränkt.

Es kann vorkommen, dass sich eine *Anforderung und deren Umsetzung über mehrere Ebenen der Meta-Modell-Hierarchie erstrecken*. Hierarchische Prozesse können nur dann modelliert werden, wenn dies die Modellierungssprache ermöglicht; solch eine Anforderung muss auf M2 definiert werden; eine Hierarchie der Prozesse an sich kann aber erst während des Modellierungsvorgangs auf M1 erstellt werden. Gleichzeitig gilt es zu berücksichtigen, dass sich die *Umsetzung der Anforderungen nicht immer nur auf die Phase des Designs beschränken muss bzw. kann*. Wie auch bereits in den Beispielen in Kapitel 4 aufgezeigt wurde, können manche Anforderungen erst im Rahmen der Modellierung realisiert werden. Beispielhaft zu nennen wäre die Anforderung, dass ein Prozessblatt nur so viele Elemente abbilden soll, wie der Anwender auf einen Blick erfassen kann. Wird daher in der Designphase auf M1 festgelegt, dass ein Prozessblatt nie mehr als sieben Prozessschritte enthalten darf, kann dies erst bei dem eigentlichen Modellierungsvorgang, der ebenfalls M1 zuzuordnen ist, umgesetzt werden.

Nichtsdestotrotz soll es dem Anwender zugestanden werden, dass ihm gewisse gute Gestaltungsmöglichkeiten erst bei der Modellierung in den Sinn kommen, weil er sich dann weniger abstrakt und theoretisch, sondern praktisch mit der Sache beschäftigen kann. Dies führt zu Iterationsschleifen in dem Gesamtprozess, die sich jedoch bekanntlich positiv auf die Qualität des finalen Ergebnisses auswirken. Anforderungen, die erst nach der anfänglich durchgeführten Anforderungsanalyse auftreten, sind in diesem Konzept nicht berücksichtigt und sind auch nicht Schwerpunkt dieser Arbeit.

Folgendes Beispiel dient der Veranschaulichung: Die Phasen in einem Prozessmodell sollen ohne Applikationen, d.h. ohne den operationalen Aspekt modelliert werden. Es ist nun zu unterscheiden, ob diese Eigenschaft in der zu verwendenden Prozessmodellierungssprache an sich berücksichtigt werden soll, oder ob lediglich eine Reihe von Prozessmodellen ohne den operationalen Aspekt modelliert werden soll.

- Im ersten Fall muss diese Anforderung auf der Ebene M2 definiert werden, das heißt die zu verwendende Prozessmodellierungssprache verzichtet explizit auf die Definition des operationalen Aspekts und eines Modellierungselements. So ist es zum Beispiel bei AOPM für die Phasen definiert.
- Im zweiten Fall wird diese Anforderung auf Ebene M1 realisiert, das heißt bei den relevanten Prozessmodellen wird kein operationaler Aspekt modelliert, obwohl dies die Sprache per Definition erlauben würde bzw. vorsieht.

5.4 Das Vorgehensmodell

Bei dem entwickelten Ansatz handelt es sich um ein *Phasenmodell*. Phasenmodelle werden generell dazu verwendet größere Projekte in überschaubare Abschnitte, d.h. Phasen zu untergliedern [Burg02] [CKKP06] [Köhl06] [DeIn09]. Laut DIN 69901 ist eine Projektphase

ein „zeitlicher Abschnitt eines Projektablaufs, der sachlich gegenüber anderen Abschnitten getrennt ist“ [DeIn09].

Design und Modellierung sind Phasen des Prozesslebenszyklus (siehe Kapitel 2.3). Darüber hinaus können die einzelnen *Ebenen der Meta-Modell-Hierarchie* M3 bis M0 den Phasen des Prozesslebenszyklus zugeordnet werden. In Kombination mit dem für Design und Modellierung entwickelten methodischen Rahmenwerk bedeutet dies: jede dieser als Phase interpretierte Ebene M3 bis M0 wird durch die (Teil-) Phasen Design und Modellierung konkretisiert bzw. umgekehrt werden Phasen Design und Modellierung durch die einzelnen Ebenen in Teilphasen untergliedert. Damit lässt sich ein Vorgehensmodell definieren, wie es in Abbildung 5-4 zu sehen ist.

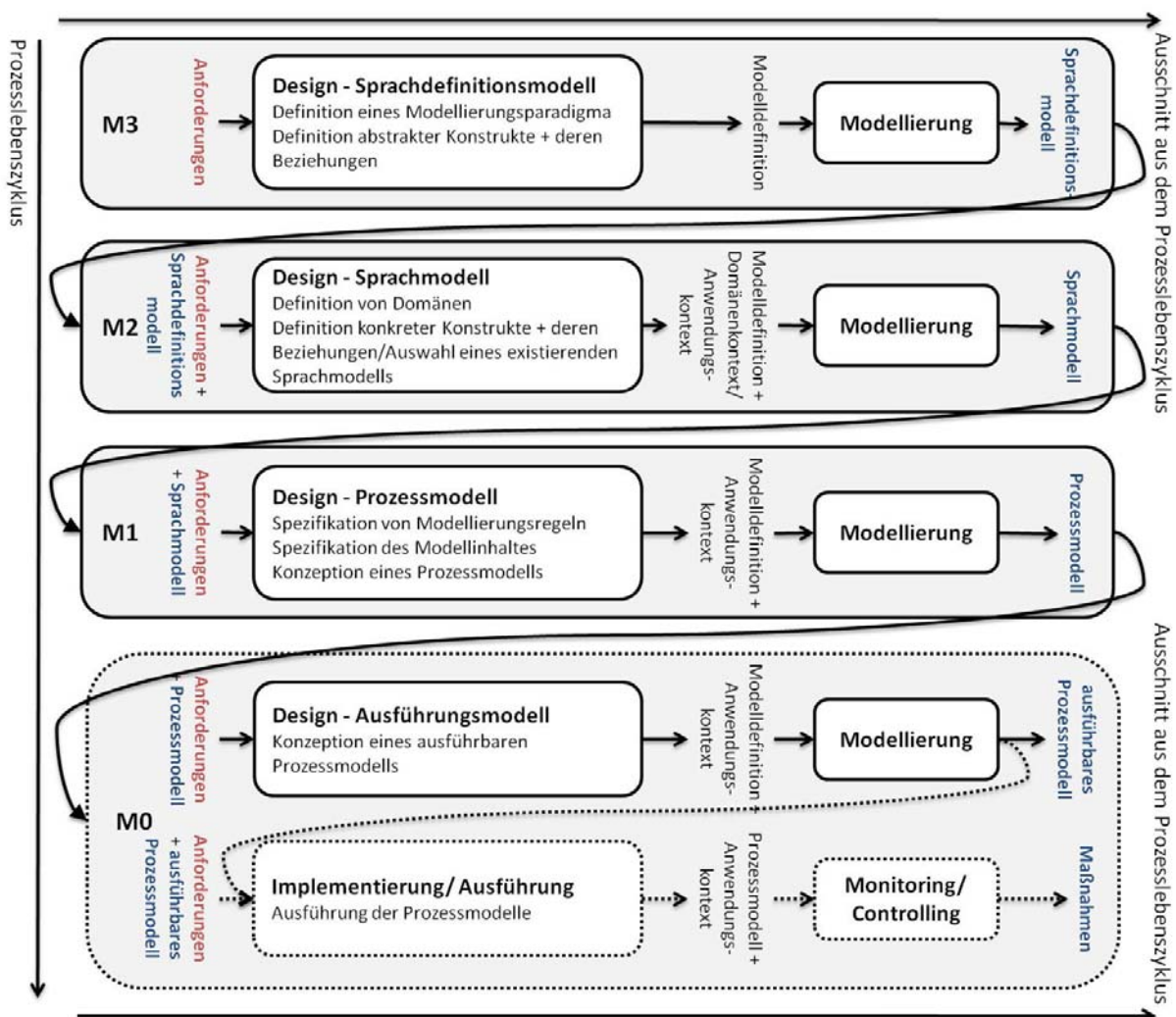


Abbildung 5-4 Vorgehensmodell für Design und Modellierung

Abbildung 5-4 gibt als Zusammenfassung Aufgaben und Inhalte von Design und Modellierung auf den verschiedenen Ebenen der Meta-Modell-Hierarchie wieder. Die detaillierten Ausführungen folgen in Abschnitt 5.5.

Von der Grundidee der zugrunde gelegten Meta-Modell-Hierarchie wird das Vorgehensmodell von oben nach unten, d.h. von M3 nach M0, durchlaufen, wie es in Abbildung 5-4 durch den Verlauf der Pfeile angezeigt wird. Der Einstieg ist jedoch, je nach Anwendungsfall, auf jeder Ebene möglich. Zudem wird der Gesamtprozess nicht so linear ablaufen, wie es die Abbildung vermuten lässt. Vielmehr wird der Ablauf von vielen ebenen- sowie phasenübergreifenden Sprüngen geprägt sein. Dies ist für die Qualität des Endergebnisses unabdingbar, da oftmals erst die Iterationen zum Optimum führen.

Vorgehensmodell Design

Im Rahmen der Anforderungsanalyse zu Beginn der Designphase wird zunächst das so genannte *Lastenheft* definiert. Das *Lastenheft* (= Anforderungsspezifikation, Anforderungskatalog) beschreibt die Gesamtheit der Anforderungen an ein zu erstellendes Objekt und dient der Klärung und Präzisierung der Aufgabenstellung [Pahl70] [Pahl74] [VDI93] [DeIn09]. Eine Anforderung wäre zum Beispiel die Möglichkeit der Unterscheidung zwischen medizinischem und administrativem Prozess. Hier erfolgt zunächst keine direkte Zuordnung der Anforderungen zu einer der Ebenen des Vorgehensmodells in Abbildung 5-4. Das zu erstellende Objekt ist hier die Prozessmodellierungssprache (inklusive Modellierungswerkzeug) sowie das Prozessmodell.

Aufbauend auf dem Lastenheft wird das *Pflichtenheft* erstellt. Dieses beschreibt in konkreterer Form die Umsetzung des Lastenhefts [DeIn09], d.h. wie und womit die Anforderungen im Lastenheft zu lösen sind. Dabei werden nun explizit die verschiedenen Ebenen des Vorgehensmodells berücksichtigt. Somit ist im Grund pro Ebene ein Pflichtenheft zu erstellen. Zum Beispiel wird auf M2 festgelegt, dass zur Unterscheidung administrative Prozesse oval, medizinische Prozessen rechteckig dargestellt werden. Zusammen mit dem jeweiligen Anwendungskontext bildet das Pflichtenheft den Input für die anschließende Modellierung.

Vorgehensmodell – Modellierung

Es erfolgt schließlich in der Phase der Modellierung die Modellerstellung. Dabei wird das umgesetzt, was im Design mittels Lasten- und Pflichtenheft identifiziert und festgelegt worden ist. Die Ergebnisse werden in den Ebenen-entsprechenden Modellen festgehalten. Nach der initialen Anforderungsidentifikation stellt nun die Identifikation der Modellinhalte aus dem aktuell vorliegenden Anwendungsfall eine Herausforderung dar. Dies ist sehr zeitintensiv und setzt gute analytische Fähigkeiten voraus. Auch der Abbildungsvorgang an sich muss sehr sorgfältig durchgeführt werden.

5.5 Erläuterungen der einzelnen Ebenen

Im Folgenden werden die Aufgaben und Inhalte der einzelnen Ebenen, getrennt nach Design und Modellierung, erläutert. Es wird auch kurz auf die unterste Ebene M0 mit der Ausführung sowie dem Monitoring/ Controlling eingegangen.

5.5.1 M3 – Design und Modellierung

Auf M3 gilt es abstrakte Modellierungsprimitive für das grundlegende Modellierungs- und Sprachparadigma zu definieren. Das Ergebnis wird in einem *Sprachdefinitionsmodell* festgehalten.

Input – Anforderungen M3

Die auf dieser Ebene zu berücksichtigenden *Anforderungen sind von generischer Natur*. Sie sind auch losgelöst von dem Prozessgedanken, der den Schwerpunkt dieser Arbeit ausmacht. Es besteht kein Bezug zu einer Anwendungsdomäne oder einem Anwendungsfall wie in Kapitel 2.2 beschrieben.

Design M3

Für die Modellierungssprache werden zunächst in abstrakter sowie generischer Form die grundlegenden *Modellierungsprimitive* spezifiziert. Werden als Modellierungs-Paradigma gerichtete Graphen definiert, sind dies zum Beispiel Knoten und Kanten. Die Anzahl der definierten Modellierungsprimitive ist gering. Erst auf den darunter liegenden Ebenen werden für jedes dieser Modellierungsprimitive mehrere konkrete Konstrukte abgeleitet.

Zur Festlegung der Struktur der Modellierungselemente, inklusive deren Beziehungen zueinander, wird eine *abstrakte Syntax* definiert [KrRV07]. Zum Beispiel wird festgelegt, dass Knoten durch Kanten verbunden werden oder dass eine Schachtelung von Knoten möglich ist. Die Definition erfolgt auf abstraktem Niveau und “describes the vocabulary of concepts provided by the language and how they are combined to create models” [CISW08].

Da ein konkreter Anwendungsbezug auf M3 fehlt, werden Aspekte der *Semantik* kaum spezifiziert, sondern nur abstrakt im Sinne von Knoten und Kanten. Damit besteht jedoch die Flexibilität einen Knoten auf einer der unteren Ebene in der einen Modellierungssprache (zum Beispiel in AOPM) als Prozessschritt zu interpretieren, in einer anderen hingegen (zum Beispiel in einem ER-Diagramm) als Beziehungstyp. Beide Modellierungssprachen können dann aber auf dieselben Grundelemente zurückgeführt werden, was die Vergleichbarkeit erhöht.

Angaben zur visuellen Darstellung werden nicht gemacht.

In enger Wechselwirkung mit der Definition der Modellierungsmethode werden auch die ersten, noch abstrakten Konzepte eines Modellierungswerkzeuges entworfen. Auf diesen Aspekt soll jedoch im Folgenden nicht näher eingegangen werden.

Modellierung M3

Die in der Designphase definierten Konstrukte können unter Berücksichtigung der definierten Syntax in einem Modell konkretisiert, d.h. modelliert werden. Damit erfolgt die Definition eines abstrakten, syntaktischen Meta-Meta-Modells, eines sog. *Sprachdefinitionsmodells*.

Der Begriff „abstrakt“ soll nicht im Sinne der Informatik verstanden werden, wo es um das Ableiten eines wesentlichen bzw. charakteristischen Aspekts aus einer Menge von Einzelobjekten geht. Vielmehr ist der Begriff in einem allgemeineren Sinn zu verstehen, nachdem die „Gegenstände nur dem Denken, nicht aber der (sinnlichen) Wahrnehmung zugänglich sind“ [Mitt05]. Mit Gegenständen sind hier die Modellierungsprimitive gemeint. Das Modell wird explizit als syntaktisch bezeichnet, da eine Semantik kaum (lediglich im Sinn von Knoten und Kanten) spezifiziert ist. Diese Konkretisierung beinhaltet prinzipiell keine visuelle Darstellung, auch wenn mit Knoten und Kanten durchaus bereits Darstellungsformen assoziiert werden können.

5.5.2 M2 – Design und Modellierung

Das Ergebnis dieser Ebene ist das Modell einer Prozessmodellierungssprache, kurz *Sprachmodell* genannt. Dabei müssen sprachbezogene Anforderungen berücksichtigt werden.

Input – Anforderungen M2

Das *Sprachdefinitionsmodell* von M3 ist als wesentlicher Input dieser Ebene anzusehen. Für die Weiterentwicklung werden hier nun sowohl weitere *generische als auch bereits domänenspezifische bzw. anwendungsbezogene Anforderungen* berücksichtigt.

Die *generischen Anforderungen* beziehen sich allgemein auf den (die) späteren Anwendungsbereich(e) der zu entwickelnden Modellierungssprache. Die *domänenspezifischen Anforderungen* beziehen sich auf spezielle Aspekte einzelner Anwendungsbereiche, wie zum Beispiel erst einmal die Domäne der Prozesse an sich, konkret die der Medizin oder der Produktentwicklung (siehe Kapitel 2.2). Es kann aber auch bereits um ein spezielles Modellierungsprojekt einer Organisation gehen, die dann *anwendungsbezogene Anforderungen* definiert. Die Anforderungen spiegeln wieder, welche Eigenschaften und Aspekte (zum Beispiel Daten und Rollen) Bestandteil des Sprachmodells sein müssen.

Design M2

Zunächst fließen die generischen Anforderungen in eine abstrakt zu definierende Zwischensprache ein (siehe Kapitel 3.3.2), die jedoch nicht für die Erstellung der Prozessmodelle auf M1 verwendet werden kann. Dazu erfolgt die Definition einer domänenspezifischen Modellierungssprache, die diejenigen domänenspezifischen Anforderungen umsetzt, die für den weiteren Verlauf in dem Vorgehensmodell von Interesse sind. Dabei muss nicht immer eine Neudefinition erfolgen, ebenso können existierende Modellierungssprachen angepasst werden. Es wird unterschieden zwischen Sprachen zur Modellierung von Geschäftsprozessen (Prozesse, bei denen die Darstellung betriebswirtschaftlich-fachlicher Aspekte im Vordergrund steht [FrvL03] [Maur96]) oder aber Workflows (Prozesse, die automatisiert ausgeführt werden oder zumindest in Teilen informationstechnisch unterstützt werden sollen [FrvL03] [Maur96]). Bei Letzteren ist eine formale bzw. exakte Definition der Modellierungssprachen unabdingbar [JaBu96] [KBGS11].

Basierend auf den Modellierungsprimitiven von M3 werden konkrete *Konstrukte* definiert. Dies erfolgt inklusive der Angabe von Attributen und Parametern, die beide erst auf den noch folgenden Ebenen mit konkreten Werten und/ oder Wertebereichen versehen werden können. Für die Beziehung der Konstrukte wird eine *konkrete Syntax* spezifiziert. Darauf aufbauend wird eine *explizite Semantik* definiert [KrRV07] [HaRu04].

Im Unterschied zur Ebene M3 können hier nun für die instanziierten, domänenspezifischen Sprachelemente Angaben zur *Visualisierung* gemacht werden. In Abbildung 5-5 wird dies beispielhaft in Form des Rechtecks für den „Prozess“ und den durchgezogenen Pfeil mit gefüllter Spitze für einen „Verbindungspfeil“ gezeigt. Form, Linienart, Farbe oder andere graphische Aspekte lassen sich beliebig anpassen. Wird zum Beispiel bei einer graphischen Modellierungssprache, wie es die perspektivenorientierte Modellierung (siehe 3.3.2.2) ist, die Anforderung nach zweierlei Arten von Verbindungspfeilen zum Verbinden der Prozesselemente gestellt, nämlich Kontroll- und Datenflusspfeile, können diese als durchgezogene bzw. gestrichelte Linie modelliert werden. Bei textuellen Sprachen kann die Darstellung der Schriftzeichen bzgl. Schriftart (Arial, Times New Roman; fett, kursiv), Größe sowie Farben individuell bestimmt werden.

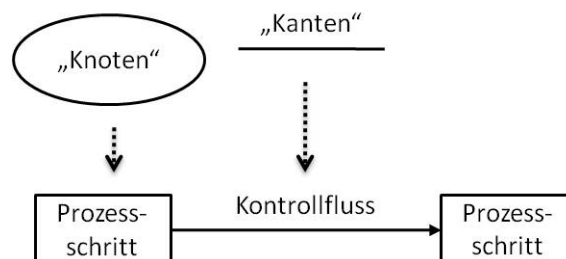


Abbildung 5-5 Beispielhafte Interpretation von "node & edge" (M3) auf der Ebene M2

Darüber hinaus wird hier auch ein Modellierungswerkzeug ausgearbeitet. Wie bereits erwähnt wird darauf im Rahmen dieser Arbeit nicht weiter eingegangen. Näheres dazu ist [Volz11] zu entnehmen.

Modellierung M2

Die im Design spezifizierte Sprache kann nun definiert werden. Dazu werden alle Konstrukte als Instanzen der Knoten und Kanten, den Modellierungsprimitiven von M3, erzeugt. Dabei werden die bereits in Kapitel 3.3.2 vorgestellten Modelltypen (APMM = abstraktes Prozess Meta Modell für die Zwischensprache, SDPMM = domänenspezifisches Prozess Meta Modell) unterschieden. Letzteres ist ein *Sprachmodell* in Form einer Prozessmodellierungssprache wie BPMN als Standardmodellierungssprache oder AOPM4MED als anwendungsbezogene Modellierungssprache, hier in Bezug auf die Medizin.

Anmerkung

Die bisher beschriebenen Ebenen M3 und M2 sind von den nun folgenden Ebenen M1 und M0 abzugrenzen. Auf M3 und M2 erfolgt die Gestaltung (= Design) sowie Konkretisierung (= Modellierung) einer Modellierungssprache mit individuellem Bezug zu einer Domäne oder einem Anwendungsfall.

Auf M1 und M0 liegen nun konkrete Anwendungsfälle vor, die mittels der auf M2 entwickelten Modellierungssprache abgebildet (M1) und dann ausgeführt (M0) werden können. Es werden keine Änderungen der zugrundeliegenden Modellierungsparadigmen und -sprachen mehr vorgenommen.

5.5.3 M1 – Design und Modellierung

Ab der Ebene M1 steht nun die Erstellung eines konkreten *Prozessmodells* im Vordergrund, das einen speziellen Anwendungsfall darstellt. Dabei wird die auf M2 definierte Modellierungssprache bzw. das *Sprachmodell* verwendet.

Input – Anforderungen M1

Als Input ist zum einen das *Sprachmodell* relevant, sei es das einer Standardmodellierungssprache oder das einer anwendungsbezogenen Modellierungssprache.

Zudem gilt es weitere *anwendungsbezogene Anforderungen* zu berücksichtigen, die sich aus dem konkreten Anwendungsfall ableiten (eine bestimmte Organisation und ihre Prozesse oder ein Klinikum und seine klinischen Pfade).

Design M1

Auf dieser Ebene geht es im Design nun nicht mehr darum, konkrete Konstrukte, wie zum Beispiel einen Prozessschritt oder Kontrollfluss an sich und deren prinzipiellen Beziehungen zu den anderen Konstrukten, zu spezifizieren. Dieser Aufgabenteil ist auf der darüber liegenden Ebene M2 abgeschlossen worden.

Der Schwerpunkt liegt unter anderem auf der *Formulierung von Regeln*, die die Verwendung der Elemente sowie das Layout des Modells betreffen. Beispielsweise kann für ein bestimmtes Projekt entschieden werden, dass der organisatorische Aspekt nicht bei kompositen Prozessen verwendet werden soll, sondern nur bei elementaren Prozessen. Ein weiteres Beispiel wäre die Regel, nie mehr als sieben Sub-Prozesse für einen Prozess und somit ein Prozessblatt zu erlauben. Diese Angaben gelten nicht allgemein in Bezug auf die Modellierungssprache, sondern nur für den aktuell vorliegenden Anwendungsfall.

Modellierung M1

Es erfolgt nun unter Verwendung einer Prozessmodellierungssprache, wie zum Beispiel AOPM oder UML, und unter Berücksichtigung der Designvorgaben, die Erstellung von **Prozessmodellen**. Es werden hier kurz grundlegende bzw. mögliche Modellarten vorgestellt, um ein Bild zu vermitteln, wie die Modellierungssprache verwendet werden kann.

- **Soll-Ist-Modelle**

Ist-Modelle werden zunächst im Rahmen einer erstmaligen Dokumentation der Prozesse erstellt und geben den aktuellen Status Quo der Abläufe, so wie sie aktuell ausgeführt werden, wieder [Fisc06] [FrRH10] [ScSe08].

Auf der Basis einer eingehenden Analyse, werden Schwachstellen, Fehler und Unstimmigkeiten aufgedeckt. Die Analyse führt zu der Definition von sog. *Soll-Modellen*, um Schwachstellen und Fehler zu eliminieren. Diese dienen dann als Basis für die Ausführung der Prozesse auf der Ebene M0 [Fisc06] [FrRH10] [ScSe08].

Die Prozessmodelle gilt es durch regelmäßige Überarbeitung einer Qualitätskontrolle zu unterwerfen, was zu einem Kreislauf der kontinuierlichen Verbesserung führt.

- **Strategisch-operative Modelle**

Strategische Modelle werden zur Darstellung des Portfolios einer Organisation eingesetzt. Dazu erfolgt die Abbildung der Hauptprozesse des Unternehmens in einer sog. Prozesslandschaft [FrRH10] [Remu02] [ScSe08] [Wilh07]. Als Beispiel wären Beschaffungs-, Produktentwicklungs-, sowie Vertriebsprozesse aus dem betriebswirtschaftlichen/ ingenieurwissenschaftlichen Bereich zu nennen.

In den *operativen Modellen* werden die Prozesse nun so detailliert abgebildet, dass die Prozesse mit den darin enthaltenen Informationen auf der Ebenen M0 mit oder ohne Systemunterstützung ausgeführt werden können. [Fisc06] [FrRH10] [JaBu96] [ScSe08] [Wilh07]. So wird zum Beispiel in einem universitären Prozessmodell für die Phase der Durchführung eines Studiengangs der Prozess „Einschreibung

durchführen“ mit dem Inputdatum „Zeugnis“ und der Organisation „Studierender“ modelliert.

5.5.4 M0 – Design und Modellierung

Es erfolgt hier nun ein Wechsel von der Modellierungsumgebung zur Ausführungsumgebung. Die *Prozessmodelle werden zunächst instanziiert*. Daher können auch auf dieser Ebene in geringem Ausmaß noch Aufgaben durchgeführt werden, die dem Design und der Modellierung zuzuordnen sind.

Input – Anforderungen M0

Bei der Instanziierung müssen *anwendungsbezogene Anforderungen* in Bezug auf das bei der späteren Ausführung zu verwendende *Prozessmodell* berücksichtigt werden. Es handelt sich hier um *planbare Anforderungen*, die die Kontextbedingungen konkretisieren und noch vor der Ausführung entsprechend in das Modell integrierbar sind bzw. integriert werden müssen. Grundlagen liefern üblicherweise die Projektpläne der Organisationen wie Termin-, Kosten- und Kapazitätspläne.

Design M0

Das Prozessmodell muss aus *organisatorischen sowie implementierungstechnischen Gesichtspunkten* an die nun konkret vorliegenden Ausführungsbedingungen angepasst werden, um ein ausführbares Prozessmodell zu erhalten. Dazu werden Parameter und Attribute mit konkreten Werten besetzt. Es kann sich auch um rein organisatorische Aspekte handeln, wie zum Beispiel dass die Mitarbeiter von ihrem Vorgesetzten ihren Aufgabenbereich zugewiesen bekommen. Ebenso kann im Hinblick auf die (informations-) technische Unterstützung der Prozessausführung festgelegt werden, dass nicht mehr als fünf Schritte pro Zeiteinheit gleichzeitig ausgeführt werden dürfen.

Eine Ausführungsumgebung für einen konkreten Anwendungsfall/ ein Projekt in einer Organisation wird entsprechend definiert bzw. im Hinblick auf die Ausführung konfiguriert. Dabei muss es sich nicht zwingend um ein IT-System handeln; es kann auch rein organisatorische Aspekte betreffen. Wie bei den Modellierungswerkzeugen soll auch darauf hier nicht näher eingegangen werden.

Modellierung M0

Die im Rahmen des Designs konkretisierten Angaben werden bei der Instanziierung in das Prozessmodell übernommen. Die Instanziierung des Prozessmodells beinhaltet die Einbettung in die technische und/ oder organisatorische Ausführungsumgebung. Dies kann sich von der web- oder papierbasierten Veröffentlichung der Prozessmodelle als Anleitung für die Mitarbeiter, über die ein teilautomatisierendes Prozessunterstützungssystem bis hin zu einem

Workflow-Management-System [JaBS99] [JaBu96] erstrecken. Damit kann dann die Prozessausführung begonnen werden.

5.5.5 M0 – Ausführung und Monitoring/ Controlling

Die hier auszuführenden Aufgaben sind den Aufgaben des operativen Projektmanagements [Birk06] [Burg02] [Ehrl07] zu zuordnen. Es erfolgt eine Detaillierung bzw. Anpassung der Planungsdaten an die aktuell vorliegenden Kontextbedingungen.

Input – Anforderungen M0

Es gilt Anforderungen zu berücksichtigen, die erst bei der Ausführung konkret zu spezifizieren sind, im Bezug auf das Vorhandensein an sich aber *planbare Anforderungen* darstellen, wie zum Beispiel die genaue Marktnachfrage nach einem Produkt oder der genaue Gesundheitszustand eines Patienten.

Anforderungen, die sich in der aktuellen Situation der Ausführung ergeben, sogenannte *unplanmäßige Anforderungen*, wie zum Beispiel der Ausfall einer Ressource, müssen nun berücksichtigt werden. Diese Art von Anforderungen stellt eine besondere Herausforderung an die Umsetzung dar, da sie nicht vorab in einem Prozessmodell festgehalten werden können.

Ausführung M0

Unter Berücksichtigung der Ausführungsbedingungen und -regeln erfolgt die Ausführung der Prozessmodelle mit oder ohne Systemunterstützung. Da die Ausführung von Prozessen nicht Schwerpunkt dieser Arbeit ist, wird hier nicht näher darauf eingegangen.

Monitoring/ Controlling M0

Das *Monitoring* erfolgt bereits während der Ausführung. Soll- und Ist-Zustand der Prozesse werden verglichen. Bei gravierenden Abweichungen lassen sich Anforderungen ableiten, die sich in kurzfristig notwendige Maßnahmen umsetzen lassen, damit der Prozess noch erfolgreich beendet werden kann. Ein mögliches Szenario wäre, dass aktuelle Leistungsanforderungen von der vorhandenen Ausführungsumgebung nicht erfüllt werden können. Die Anforderung wäre diese Leistungslücke zu schließen. Als Reaktion könnte zum Beispiel der Server-Rechner ausgetauscht werden. Andernfalls, zur Änderung der Prozessmodelle und/oder der Prozessmodellierungssprache, werden die Maßnahmen auf den Start eines der nächsten Prozesslebenszyklen verschoben.

Das *Controlling* wird erst nach der Ausführung eines Prozesses durchgeführt. Hier liegt der Schwerpunkt auf dem Aufdecken von Verbesserungspotential. Daraus ergeben sich neue Anforderungen im Hinblick auf den nächsten Prozesslebenszyklus. Mit dem erneuten Start einer Designphase erfolgt der Übergang zur kontinuierlichen Prozessverbesserung in Form

eines Kreislaufs, wie sie in Kapitel 2.3.1 in Form des Kreislaufs in Abbildung 2-4 aufgezeigt wurde.

5.6 Prototypische Realisierung in OMME

Im Hinblick auf die Möglichkeit eine (Prozess-) Modellierungssprache individuell für eine Domäne oder einen speziellen Anwendungsfall entwickeln oder anpassen zu können, wird an dieser Stelle die am Lehrstuhl entwickelte *Open Meta Modeling Environment (OMME)* [Volz11] vorgestellt; sie wird auch noch im weiteren Verlauf der Arbeit relevant sein (siehe Kapitel 6.5.8). OMME ist eine generische Plattform zur freien Entwicklung, Bearbeitung und Visualisierung von Meta-Modellen, das heißt Modellierungssprachen können individuell definiert werden [Volz11] [VoJa10a] [VoJa10b]. Sie ermöglicht damit auch die Erstellung von Anwendermodellen, wie zum Beispiel ein AOPM-Prozessmodell. OMME ist somit ein Beispiel, das zeigt, wie Anwender mit Hilfe einer generischen Werkzeugunterstützung individuell ihre eigene (Prozess-) Modellierungssprache flexibel gestalten können. Im Folgenden werden Konzepte und Architektur von OMME erläutert. Zum besseren Verständnis erfolgt die Darstellung an Hand eines Use Cases.

5.6.1 Konzept und Architektur von OMME

Implementierungstechnisch besteht das Modellierungswerkzeug OMME aus einer Reihe von *Plug-ins für die Eclipse Plattform*. Eclipse [EcF11] ist ein Programmierwerkzeug zur Entwicklung von Software verschiedenster Art. Darüber hinaus werden verschiedene bereits vorhandene Frameworks verwendet, wie zum Beispiel EMF [ThEF11a] für die Speicherung von Modellen, Xtext [ThEF11c] und GMF [ThEF11b] für einen textuellen bzw. graphischen Editor.

Mit OMME können beliebig viele Meta-Hierarchien und die dazugehörigen Modelle erstellt und gespeichert werden. Ein Beispiel ist die in 3.3.3 vorgestellte Meta-Modell-Hierarchie. Es können in OMME Änderungen auf allen Hierarchie-Ebenen eingefügt werden, so dass auch völlig neue Modellierungskonzepte definiert werden können.

Die Definitionen zu den Modellen der einzelnen Ebenen werden in einem sog. *linguistischen Meta Modell (LMM)* abgelegt (siehe Abbildung 5-6, linke Seite). Es erlaubt beliebig viele Hierarchien abzubilden, die sich hier in den Teilmodellen zeigen. Auf das linguistische Meta Modell soll im Rahmen dieser Arbeit nicht näher eingegangen werden, der interessierte Leser sei auf [Volz11] verwiesen. Entscheidend für diese Arbeit ist es, dass dieses Modell die Grundlage für OMME bildet, in der die wesentlichen Aspekte gespeichert werden.

In die Modellierungsumgebung ist auch eine entsprechende *Konsistenzprüfung* integriert. Sobald also Änderungen auf einer Ebene Fehler auf einer anderen Ebene zur Folge haben, wird dies im Werkzeug entsprechend angezeigt.

Ist unter Verwendung dieses Werkzeuges eine Modellierungssprache definiert, wird diese schließlich auf ein graphisches Modell abgebildet. Damit wird ein *graphischer Editor* generiert, der die Schnittstelle zum Benutzer, dem eigentlichen Modellierer, darstellt. Der Editor wird von Stakeholder zur Erstellung der Anwender-Prozessmodelle verwendet, wie es in Abbildung 5-6 am Beispiel eines AOPM-Prozessmodells zur Darstellung eines Teilprozesses aus dem medizinischen Projektes des Lehrstuhls zu sehen ist.

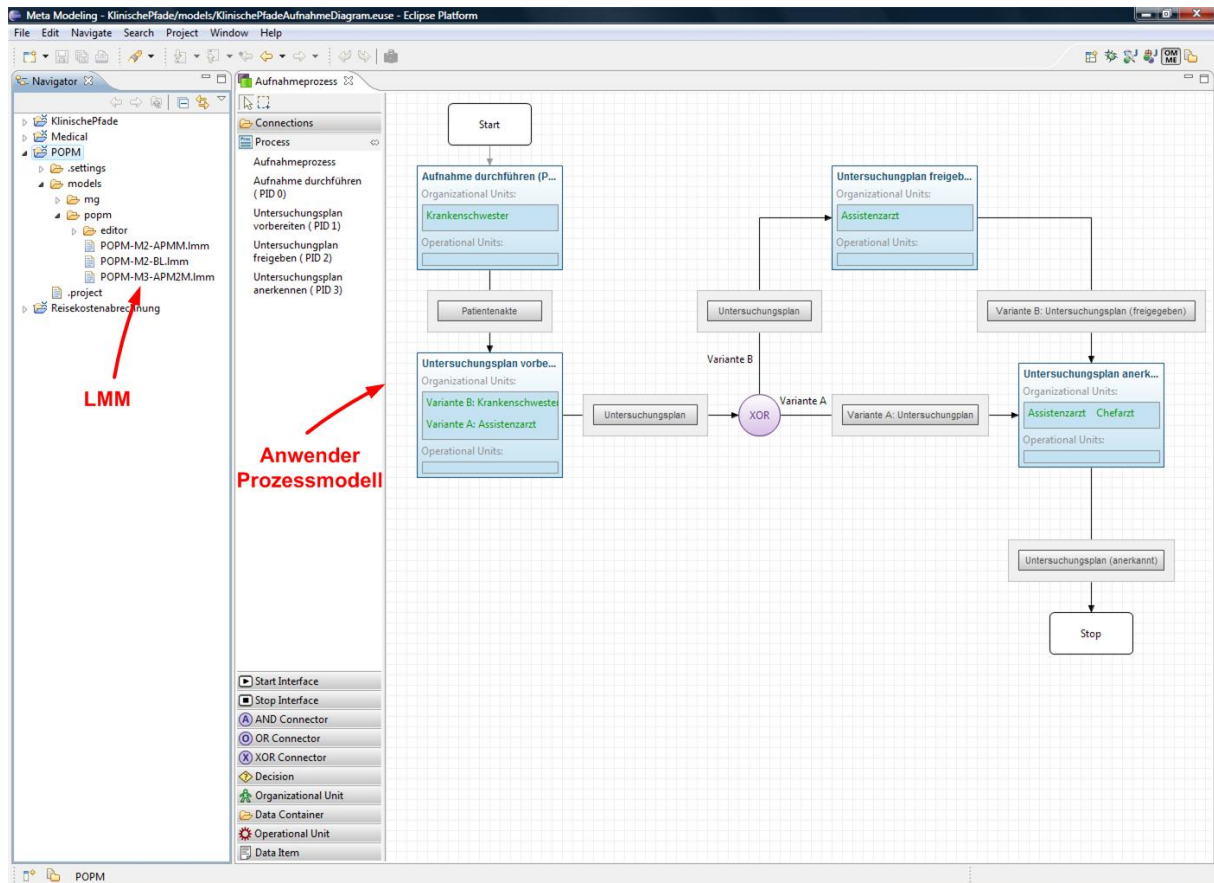


Abbildung 5-6 Screenshot OMME [eigene Quellen]

5.6.2 Use Case

Als Use Case für den generischen Prototyp wurden bisher Meta-Modelle für die Perspektivenorientierte Modellierung sowie ER-Diagramme erstellt (siehe [Volz11]). Ersteres ist im Kontext dieser Arbeit von Interesse, daher wird näher darauf eingegangen.

Das Meta-Modell für AOPM wird über drei Hierarchieebenen (M3, M2 und M1) definiert.

- Die oberste *Ebene M3* beschreibt allgemein bzw. generisch, wie die Modellierungssprache konzeptionell aufgebaut ist. Es handelt sich bei AOPM um einen gerichteten Graphen, der Zyklen enthalten kann. Als grundlegende Konzepte

werden die für die Prozessmodelle relevanten Perspektiven, wie zum Beispiel die Daten-orientierte oder organisatorische Perspektive, sowie Knoten (= Node) definiert. In Abbildung 5-7 ist ein Ausschnitt aus dem Quellcode des zu AOPM gehörenden LLM angezeigt. Dieser beschreibt die Deklaration eines Knoten als Prozess mit den für diese Modellierungselemente relevanten Perspektiven. An dieser Stelle wird nicht näher auf einzelne Aspekte eingegangen, da ein tieferes Verständnis im Rahmen dieser Arbeit nicht verlangt wird. Der interessierte Leser sei auch an dieser Stelle auf [Volz11] verwiesen.

```

1  ...
2  concept Node {
3  attributes {
4
5  concept BehavioralPerspective inboundControlFlows { multiplicity = zeroOrMore; }
6  concept BehavioralPerspective outboundControlFlows { multiplicity = zeroOrMore; }
7
8  string function { multiplicity = zeroOrOne; }
9  concept OrganisationalPerspective organisations { multiplicity = zeroOrMore; }
10 concept OperationalPerspective operations { multiplicity = zeroOrMore; }
11
12 concept DataFlow inboundDataFlows { multiplicity = zeroOrMore; opposite = target; }
13 concept DataFlow outboundDataFlows { multiplicity = zeroOrMore; opposite = source; }
14
15 concept Node innerContent { multiplicity = zeroOrMore; }
16
17 }
18
19 }
20 ...

```

Abbildung 5-7 Deklaration eines Knoten (= Node) im LLM - M3

- Auf *Ebene M2* wird zunächst eine abstrakte Prozessmodellierungssprache, die Zwischensprache, definiert. Darauf aufbauend erfolgt die Definition der konkreten und verwendbaren Prozessmodellierungssprache AOPM. Daher werden hier auch zwei verschiedene LMM-Modelle erstellt: ein allgemeines mit dem Suffix BL, sowie ein domänenspezifisches mit dem Suffix APMM (siehe Abbildung 5-6).
- Auf der untersten *Ebene M1* wird schließlich das eigentliche Prozessmodell erstellt, wie es zur Ansicht für den Anwender verwendet wird. Hier kommt schließlich auch der graphische Editor zum Einsatz wie er in Abbildung 5-6 bereits mit einem einfachen POPM-Prozessmodell zu sehen war.

Es muss betont werden, dass es sich dabei um einen Prototyp handelt, der derzeit durch Modellierung diverser Anwendungsfälle evaluiert wird.

5.7 Zusammenfassung

Die Ebenen des hier vorgestellten Vorgehensmodells werden nicht alle in gleichem Ausmaß genutzt:

Der Einstieg auf *Ebene M3* im Rahmen der vorgestellten und erweiterten Meta-Modell-Hierarchie erfolgt selten bis gar nicht; erst wenn sämtliche Gestaltungsmöglichkeiten zur Umsetzung von Anforderungen auf den unteren Ebenen ausgeschöpft sind oder sich als nicht adäquat erwiesen haben. Das geltende Modellierungsparadigma scheint dann prinzipiell nicht (mehr) ausreichend zu sein.

Wenn zu erkennen ist, dass die verfügbaren sprachlichen Mittel, d.h. die Funktionen, Eigenschaften sowie inhaltliche Bedeutung einer verfügbaren Modellierungssprache nicht optimal sind, sind Veränderungen des Sprachmodells, das heißt der Modellierungssprache, auf der *Ebene M2* vorzunehmen. Ursache kann eine abweichende, geänderte oder erweiterte Betrachtungsweise des Sachverhalts sein. Beispielsweise werden in einem Unternehmen neue Kennzahlen definiert, die im Rahmen des Monitorings und Controllings analysiert werden müssen. Nicht mehr notwendige Kennzahlen sollten dann gelöscht werden, um sich auf die wesentlichen Aspekte konzentrieren zu können. Änderungen an einer Modellierungssprache sind auch dann nötig, wenn sich ihr Einsatzgebiet verschiebt, zum Beispiel von einem rein betriebswirtschaftlichen Fokus hin zur Medizin, die zum Beispiel im Bezug auf Kennzahlen völlig andere Aspekte betrachtet. Weitere Aspekte sind bereits in Kapitel 4 betrachtet worden.

Bei als dauerhaft anzusehenden Änderungen bzgl. der Prozesslogik und/ oder deren Aspekte und Inhalte können Änderungen am Prozessmodell oder den Modellierungsvorschriften auf der *Ebene M1* notwendig sein. Änderungen der allgemeinen Wettbewerbssituation können Auswirkungen auf das Unternehmensportfolio haben. Geschäftsfelder und deren Prozesse müssen dann zum Beispiel gelöscht, geändert und/ oder neue hinzugefügt werden. Neue Gesetzesvorgaben, Vorschriften und Standards wirken sich ebenfalls auf dieser Ebene aus. So ergibt sich zum Beispiel die Notwendigkeit zur Angabe eines Datenflusses zusätzlich zum Kontrollfluss.

Nach der Instanziierung auf *Ebene M0* wird allgemein viel Zeit und Arbeitseinsatz darauf verwendet, auf unvorhergesehene Ereignisse, wie der Ausfall einer Ressource, zu reagieren. So etwas lässt sich nicht immer vermeiden und ist in dem Fall unabhängig von der Qualität des zugrundegelegten Prozessmodells. Dennoch kann es kritische Punkte geben, die durch eine bessere Vorbereitung, das heißt Entwicklung einer geeigneten Modellierungssprache und Erstellung eines adäquaten Prozessmodells, hätten vermieden werden können.

Zur Veranschaulichung des Vorgehensmodells soll ein Vergleich mit den Arbeiten von Österle [Öste94] vorgenommen werden. Er vergleicht das Ausmaß der Handlungen im Bereich des Business Engineering (= Neugestaltung der Wirtschaft auf Basis der Informationstechnologien [DaSh90] [KBGS11]) mit den Begriffen der Revolution sowie Evolution. Auf den *Ebenen M1 und M0* des in diesem Kapitel erarbeiteten Rahmenwerks werden keine Änderungen des zugrundeliegenden Sprachkonzeptes oder -paradigmas vorgenommen; die Prozessmodelle werden im Bezug auf einen konkreten Anwendungsfall generiert. Die beiden Ebenen können daher mit dem *evolutionären Ansatz* nach [Öste94] verglichen werden. Kommt es zu einer grundlegenden (Neu-) Entwicklung des

Sprachmodells, das heißt der Modellierungssprache, oder sogar des grundlegenden Modellierungsparadigma auf den *Ebenen M2 bzw. M3*, ist diese nach Österle [Öste94] insbesondere bzgl. der Ebene M3 vergleichbar mit einer *Revolution* (lateinisch: „radikale Veränderung der gegebenen Bedingungen“ [ScKl06]).

Während Österle den Übergang von der Revolution zur Evolution fließend darstellt, ist bei dem in dieser Arbeit entwickeltem Ansatz auf Basis der Meta-Modell-Hierarchie eine klare Strukturierung mit einer konkreten Definition von Aufgaben gegeben. Dabei sollte aber berücksichtigt werden, dass sich in einem Modellierungsprojekt die Summe der verschiedenen Maßnahmen beliebig auf die verschiedenen Ebenen verteilen lassen.

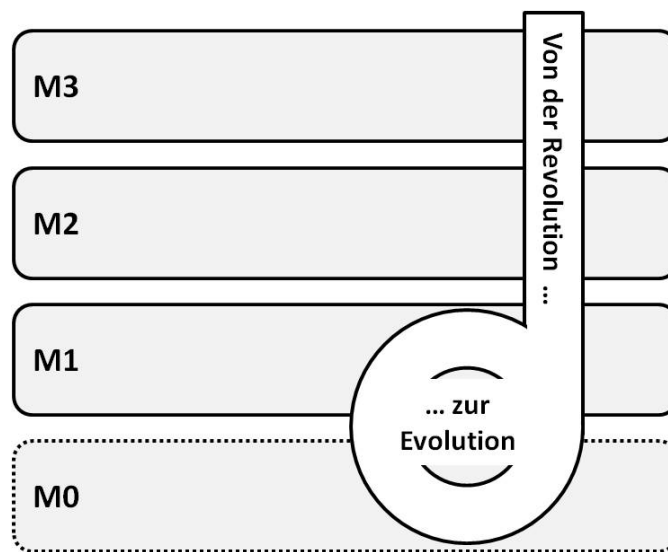


Abbildung 5-8 Von der Revolution zur Evolution

Die Ausführungen zeigen, dass es Sinn macht vor Beginn eines Prozessmanagementprojektes eine *explizite und kritische Anforderungsanalyse* durchführen, um darauf aufbauend individuelle Umsetzungsmaßnahmen ableiten zu können. Dabei kann durch explizite *Gestaltung der Prozessmodellierungssprachen* das Management der Prozesse aktiv gestaltet werden und nicht bzw. weniger reaktiv. Letzteres führt oftmals zu einer unsauberen Verwendung einer Prozessmodellierungssprache oder zusätzlichen Maßnahmen bei der Prozessausführung aufgrund nicht optimaler Prozessmodelle.

Ein Beispiel im Bezug auf Prozessvarianten soll dies noch einmal verdeutlichen. So kann eine explizite Anforderungsanalyse ergeben, dass eine Modellierungssprache mit ihren Eigenschaften nicht dazu geeignet ist, Varianten übersichtlich und kompakt in einem Modell darzustellen. Darüber hinaus wird festgestellt, dass es sinnvoll ist den Anwender beim Ableiten der individuellen Varianten aus dem integrierten Modell zu unterstützen. Demzufolge bietet es sich für den ersten Punkt an, die Möglichkeiten des Designs auf M2 (gegebenenfalls auch auf M3) zur Definition einer konfigurierbaren Modellierungssprache zu

nutzen. Diese sollte dann zum Beispiel ein spezielles Modellierungskonstrukt (variation point) zur Anzeige der variablen Stellen in den Prozessen enthalten. Die geometrische Form, die Farbe und die Zuordnung zu den gewünschten Aspekten eines Prozessmodells müssen definiert werden. Die Modellierungssprache wird darüber hinaus auf der Ebene M1 durch die Angabe spezieller Modellierungsregeln ergänzt, so dass es zum Beispiel erlaubt sein kann die variation points beim Modellierungsvorgang ineinander zu verschachteln. Zur Anwendung des Modells auf M0 wird schließlich ein Konfigurationskonzept definiert, das den Anwender bei dem Ableiten der Varianten unterstützt. Dabei kann zum Beispiel festgelegt werden, dass der Anwender die variation points unter Berücksichtigung der entsprechenden Bedingungen, in beliebiger Reihenfolge auflösen kann.

Die *Meta-Modell-Hierarchie* stellt letztendlich ein *Rahmenwerk für das Design* zur Verfügung, mit der die Aufgaben der Anforderungsanalyse sowie der Gestaltung (siehe Abbildung 1-2) strukturiert, differenziert und zielorientiert durchgeführt werden können. Es handelt sich dabei um eine allgemeine Vorgehensweise, die auf jeden beliebigen Anwendungsfall übertragbar ist.

Kapitel 6

Der Prozesskonfigurator

6.1 Motivation

“Any customer can have a car painted any colour that he wants as long as it is black” [Ford22]. Henry Fords Aussage aus dem Jahre 1914 ist inzwischen nicht mehr gültig. Heutzutage gibt es zahlreiche verschiedene Automobile, sog. *Varianten*, auf den Straßen zu sehen. Grob definiert stellen Varianten Artefakte mit ähnlicher Form und Funktion dar, die in großen Teilen übereinstimmen [DeIn02a]. Die große Zahl von Varianten, die vom Markt bzw. Kunden gefordert wird, stellt die Hersteller vor eine große Herausforderung in Entwicklung/Konstruktion und Fertigung. Ziel muss es deshalb sein, den Herstellern Methoden und Werkzeuge bereit zu stellen, mit denen sie eine Vielzahl von Varianten mit einem Minimum an Aufwand realisieren können. Das damit breite Spektrum an unterschiedlichen Produkten kann entscheidenden Wettbewerbsvorteil gegenüber den Konkurrenten darstellen.

Entsprechend der hohen Variantenzahl bei den Automobilen an sich, variieren auch die dazugehörigen Entwicklungs- und Produktionsprozesse [GoAJ08] [HaBR08b] [KiKr08] [VDA05], bis hin zum Recyclingprozess eines solchen Produktes (siehe dazu zum Beispiel [RMTF99]). Während sich dieses Beispiel primär auf den Ingenieurbereich bezieht, haben die Erfahrung des Lehrstuhls für Angewandte Informatik IV der Universität Bayreuth in diversen Projekten gezeigt, dass *Prozessvarianten* unter anderem auch in dem Bereich der Verwaltung und der Medizin zu finden sind (zu den Projekthalten siehe die Einführung zu Kapitel 4).

Varianten sind in den letzten Jahren bereits ein viel diskutierter Aspekt des *Designs* [RovA07]. Da Varianten in vielen verschiedenen Anwendungsbereichen zu finden sind, ist es sinnvoll, diesen Aspekt auch anwendungsbereichsübergreifend zu betrachten, um möglicherweise dieselben Methoden und Konzepte anwenden zu können bzw. existierende Konzepte und Methoden nach einer entsprechenden Anpassung übernehmen zu können. Aufwendige Einzellösungen können somit vermieden werden.

Eine verbreitete *Methode zum Management von Varianten* sowie eine typische Design-Aktivität stellt die *Konfiguration* dar [BAKF04] [GoAJ08] [HaBR08c] [LaRo08a] [Riit01] [TrGC95]. Dabei geht es um die Gestaltung von Artefakten basierend auf einer Menge vorab definierter Elemente [MiFr89]. Im Bereich des Prozessmanagement sind die Artefakte die Prozessmodelle und die Elemente die diversen Prozesselemente wie Daten, Prozessschritte oder Werkzeuge. Eine essentielle Grundlage aber auch Herausforderung für ein umfassendes Variantenmanagement- bzw. Konfigurationskonzept ist die Speicherung bzw. Abbildung der

Varianten in einem entsprechenden Datenmodell [BDDK04] [TiSo97] [SaWe97]. Nicht weniger herausfordernd ist es, aus dieser Datenbasis die für den Anwender passende Variante zu extrahieren, um ihm damit die von ihm gewünschten Informationen liefern zu können [HaCh93] [ReMT09] [ReRU00].

Für ein umfassendes und integriertes Management variantenreicher Prozesse ist im Rahmen der vorliegenden Forschungsarbeit das Konzept eines *Prozesskonfigurators* mit diversen Teilkomponenten entwickelt worden. Für das Ableiten der einzelnen Varianten aus den Prozessmodellen wurde ein *gestufter Konfigurationsprozess* ausgearbeitet. Ziel war es, den Anwender auf der einen Seite durch den variantenbezogenen Entscheidungsprozess zu leiten, ihn auf der anderen Seite aber auch so wenig wie möglich in seiner Vorgehensweise einzuschränken. Zur kompakten und strukturierten Abbildung der Varianten in einem Modell wurde ein *generisches, konfigurierbares Datenmodell* entwickelt. Dies basiert auf einem bereits existierenden Konzept zur Abbildung variantenreicher Strukturen in Form des sog. mereologischen Graphen. Dieser wurde an die Eigenschaften von Prozessen angepasst, wobei darauf Wert gelegt wurde, unabhängig von existierenden Prozessmodellierungssprachen wie zum Beispiel AOPM oder BPMN zu sein.

Die Ausarbeitung des Konzeptes des Prozesskonfigurators ist als ein beispielhafter Anwendungsfall für das (*Prozess-*) *Designs* zu sehen, wie es im Kapitel 3 und Kapitel 5 erläutert wurde. Zum einen ist die *Definition einer Modellierungssprache* notwendig, um die Darstellung der Varianten mittels eines generischen, konfigurierbaren Modells zu ermöglichen. Zum anderen besteht die Notwendigkeit eine *Methodik bzgl. der Verwendung des Modells*, d.h. zur Ableitung der einzelnen Varianten aus einem konfigurierbaren Prozessmodell. Beide Aufgabenstellungen sind klar dem Design zuzuordnen. Entsprechend den vorherigen Ausführungen bilden die im Folgenden zunächst beschriebene *Anforderungsanalyse*, inklusive der Einführung in das Thema (*Prozess-*) Varianten, die Grundlage für die nachfolgende *Gestaltung* des Konzeptes für den Prozesskonfigurator. Neben der Konzeption des gestuften Konfigurationsprozesses liegt der Schwerpunkt auf der Ausarbeitung des konfigurierbaren Datenmodells als Datenbasis für den Prozesskonfigurator. Dazu kann der erste Prototyp eines Modellierungswerkzeuges vorgestellt werden.

6.2 Der Begriff der Variante

Der Begriff der Variante wird allgemein eher uneinheitlich verwendet, weshalb unterschiedliche inhaltliche Bedeutungen existieren. Im Folgenden wird auf die Definition des Begriffs der Varianten eingegangen und eine für den Kontext dieser Arbeit gültige Definition herausgearbeitet. Um diese Definition noch weiter zu präzisieren wird eine Abgrenzung zu den Begriffen des Entscheidungspunktes und der Version vorgenommen.

6.2.1 Definition Variante

Die DIN 199-2 [DeIn02a] definiert den Begriff der Variante als:

... Gegenstände ähnlicher Form und/oder Funktion mit einem in der Regel hohen Anteil identischer Gruppen oder Teile...

Diese Definition legt den Schwerpunkt auf das Objekt mit seiner äußeren Gestalt (Form) und die in das Objekt integrierten Funktionen. Bzgl. der quantitativen Unterschiede zwischen den Varianten, d.h. wieviele Abweichungen auftreten müssen, werden keine konkreten Aussagen gemacht (was bei den im Rahmen dieser Arbeit analysierten Definitionen der Fall war). Auch die im Kontext dieser Arbeit untersuchten Prozessmodelle führten zu keinem anderen Ergebnis. Vielmehr sind oftmals qualitative, kontextbezogene bzw. subjektive Faktoren entscheidend. Es kann ausreichend sein, dass sich ein Element von einem anderen Element in einer Eigenschaft unterscheidet.

Definitionen aus speziellen Anwendungsbereichen, wie dem technisch-wirtschaftlichen, fallen wesentlich knapper aus. So definiert der VDI [VDRE76] Varianten lediglich als:

... Abart einer Grundaussführung ...

Mit dem Begriff Abart wird keine Aussage über die Art der Unterschiede, zum Beispiel bzgl. Form und/oder Funktion getroffen, wie es bei [DeIn02a] der Fall ist. Varianten können hier jedoch einer gemeinsamen Basis, der Grundaussführung, zugeordnet werden, ohne dass spezifiziert wird, wie diese genau zu gestalten ist. Somit kann die Grundaussführung eine Reduktion auf die übereinstimmenden notwendigen Teile, den Standardprozess oder alle Inhalte in integrierter Form darstellen [RoSc97].

Abstrakter ist die Definition, die im Bereich der Softwareentwicklung [Clem06] [PBvL05] verwendet wird. Dort werden Varianten definiert als:

... verschiedene Ausprägungen eines Produktes ...

Die Ausprägungen beziehen sich auf die Eigenschaften eines Produktes, die je nach Variante verschieden sind. Dabei wird auf inhaltliche und/oder strukturelle Aspekte des Produktes und seine Eigenschaften nicht eingegangen. Dadurch ist weitaus mehr Interpretationsspielraum gegeben als bei den vorherigen Definitionen.

Allen hier genannten Definitionen ist gemeinsam, dass die Veränderungen an den jeweiligen Objekten unabhängig vom Faktor Zeit erfolgen, d.h. ihr Vorhandsein bzw. Auftreten an sich entscheidend ist.

Die Definitionen gehen nicht konkret auf Prozessvarianten ein. Nach Analyse der aus den Projekten zur Verfügung stehenden Prozessmodelle wurde eine für diese Arbeit gültige Definition erarbeitet. Eine *Prozessvariante* wird demnach hier folgendermaßen definiert:

- Varianten eines bestimmten Prozesses erfüllen alle eine ähnliche Funktion; im übergeordneten Sinn ist es dieselbe Funktion [Gabl10]. Damit besteht unter gewissen Rahmenbedingungen, die konkret definiert werden können, eine *potentielle Austauschbarkeit* zwischen den einzelnen Varianten.
- Die Prozessvarianten unterscheiden sich *signifikant* [SAKR05] bzgl. eines oder mehrerer *Aspekte* [Ehrl07] bzw. *Prozesselemente*, an einer oder mehreren *Stellen* des Prozessmodells. Diese Aspekte, Elemente bzw. Stellen sollen im Folgenden als *variation point* bezeichnet werden. Was signifikant konkret bedeutet, ist vom jeweiligen Kontext abhängig bzw. subjektiv zu beurteilen. Die einzelnen Varianten beziehen sich immer auf einen individuellen Kontext, der sich vom Kontext einer anderen Variante unterscheidet [HaBR08a].
- Es handelt bei einem variation point um eine besondere Art eines Entscheiders bzw. Konnektors, der bzgl. der signifikanten Merkmale eines Prozessmodells entscheidend ist. Die Grundstruktur des Prozessmodells ist dadurch nicht bzw. wenig beeinflusst.
- Die Unterschiede können sich auf jeden *beliebigen Aspekt* bzw. *jedes Prozesselement* der zu vergleichenden Prozesse beziehen: Funktion, Daten, Rollen, Werkzeuge, Kontrollfluss etc..
- Die Prozesselemente können im Bezug auf den Vergleichsprozess fehlen, zusätzlich vorhanden sein, eine andere Position im Modell haben (*strukturelle Änderungen*). Die Elemente können aber ebenso bei unveränderter Position lediglich inhaltlich geändert worden sein (*inhaltliche Änderungen*). Die Unterschiede sind somit sowohl in *quantitativer als auch in qualitativer* Hinsicht möglich.
- Die Definition von Varianten kann sich auf den Gesamtprozess beziehen oder aber nur auf einen Ausschnitt aus dem Gesamtmodell.
- Die Varianten existieren in Bezug auf ihre Definition *zeitlich nebeneinander*, so dass ein entsprechendes Prozessmodell zunächst einmal mehrere Varianten integrieren kann. Im Rahmen des Prozessmanagements schließen sie sich jedoch zum Zeitpunkt der Ausführung gegeneinander aus, so dass zu dem Zeitpunkt letztendlich maximal eine Variante ausgeführt wird.

Diese perspektivenorientierte Betrachtungsweise ist besonders hervorzuheben. Sie ist für eine umfassende Definition von Prozessen unverzichtbar [JaBu96] und somit auch in Bezug auf die Prozessvarianten und deren Unterschiede zu berücksichtigen. Schließlich können sich diese laut der obigen Definition auf jeden beliebigen Aspekt eines Prozesses beziehen. Viele Ansätze im Bezug auf das Variantenmanagement konzentrieren sich jedoch lediglich auf den Kontrollfluss [LaRo08a].

Hervorzuheben ist auch die Unterscheidung in qualitativer wie quantitativer Hinsicht. Die eingehende Analyse der Lehrstuhlprojekte hat gezeigt, dass bzgl. quantitativer Unterschiede keine genaueren Angaben gemacht werden können. Letztendlich hängt die Entscheidung, ob ein Prozess eine Variante ist oder nicht, ob er sich also signifikant unterscheidet, von der subjektiven Meinung des Betrachters bzw. der Qualität der Elemente ab. In einem Fall kann bereits die Änderung von einem oder zwei Aspekten die Definition einer Variante ausmachen. So führt die Tatsache, dass es sich bei der elektiven Aufnahme eines Patienten in einem Klinikum um einen Privatpatienten handelt, für den ein extra Wahlleistungsvertrag abgeschlossen wird (was lediglich einen zusätzlichen Schritt inklusive Output-Datum im Vergleich zu dem elektiven Aufnahmeprozess eines Kassenpatienten ausmacht), zur Definition einer Variante. In einem anderen Fall würde das Hinzunehmen eines einzelnen Schrittes nicht weiter ins Gewicht fallen bzgl. der Definition von Varianten.

Entscheidend bei der Definition von Varianten ist, dass durch die gezielte Kombination von definierten Prozesselementen die wesentlichen Eigenschaften der Prozesse und die finale Struktur spezifiziert werden.

6.2.2 Abgrenzung Variante - Entscheidungspunkte

Zum besseren Verständnis der Prozessvarianten sowie des in dieser Arbeit entwickelten Konfigurationskonzeptes werden die Begriffe *Entscheidungspunkte* und *Variante* kurz einander gegenübergestellt sowie voneinander abgegrenzt.

Entscheidend sind dabei zunächst folgende Aspekte:

- *Entscheidungspunkte* beziehen sich auf die *Ausführungszeit*. Für die Auflösung eines Entscheidungspunktes sind die während der Ausführung aktuell vorliegenden *Werte* zu den zuvor im Design definierten Parametern notwendig (zum Beispiel der genaue Kaufpreis eines Produktes).
- Die Definition einer *Variante* beinhaltet Entscheidungen in Bezug auf das *Prozessdesign*. Hier gilt es den Prozess mit seine konkreten Struktur und den wesentlichen Eigenschaften zu gestalten. Die Varianten und die Ausprägung ihrer entscheidenden Merkmale leiten sich dabei ab aus *Anforderungen*, die sich aus dem vorgesehenen *Anwendungsszenario* der Prozesse ergeben (zum Beispiel Kundenwünsche, Projekt- oder Compliance-Anforderungen).

Entscheidungspunkte - Alternativen

Ein Prozess enthält üblicherweise mehrere *Entscheidungspunkte*, die in den Prozessmodellen mittels Konnektoren oder Entscheidern dargestellt werden. Ein Beispiel in Bezug auf den organisatorischen Aspekt ist in Abbildung 6-1 am PID 3 mit der Nummerierung ① zu sehen, wo die Wahl zwischen Assistenz- und Chefarzt besteht. Einem solchen Entscheidungspunkt sind stets mehrere, mindestens zwei, sog. *Alternativen* als Auswahlmöglichkeiten zugeordnet.

Die anderen Auswahlmöglichkeiten innerhalb des Prozessmodells in Abbildung 6-1 werden an dieser Stelle zunächst einmal bewusst vernachlässigt.

Entscheidend ist, dass solche Entscheidungspunkte definitiv erst während der *Prozessausführung* getroffen werden (können). Der Anwender benötigt zu der Entscheidung die dann aktuell vorliegenden *Werte*. Die Entscheidung an ① wird aufgelöst je nachdem, ob der Assistenzarzt oder der Chefarzt Dienst hat.

Ein weiterer entscheidender Aspekt ist, dass bis zu einem gewissen Grad die Entscheidungspunkte *auf den Gesamtprozess bezogen isoliert*, d.h. für sich betrachtet werden. Die Auflösung eines Entscheidungspunktes zur Ausführungszeit hat weder auf eine noch nachfolgende Entscheidung direkten Einfluss, noch müssen bereits getroffene Entscheidungen direkt berücksichtigt werden (indirekt geschieht dies natürlich unter Berücksichtigung des dann bei der Prozessausführung allgemein vorliegenden Kontextes). So wird in dem Beispiel in Abbildung 6-1⁷ die Entscheidung, ob die Alternative Assistenzarzt oder die Alternative Chefarzt den Schritt ausführt, unabhängig von dem davorliegenden Prozessablauf getroffen.

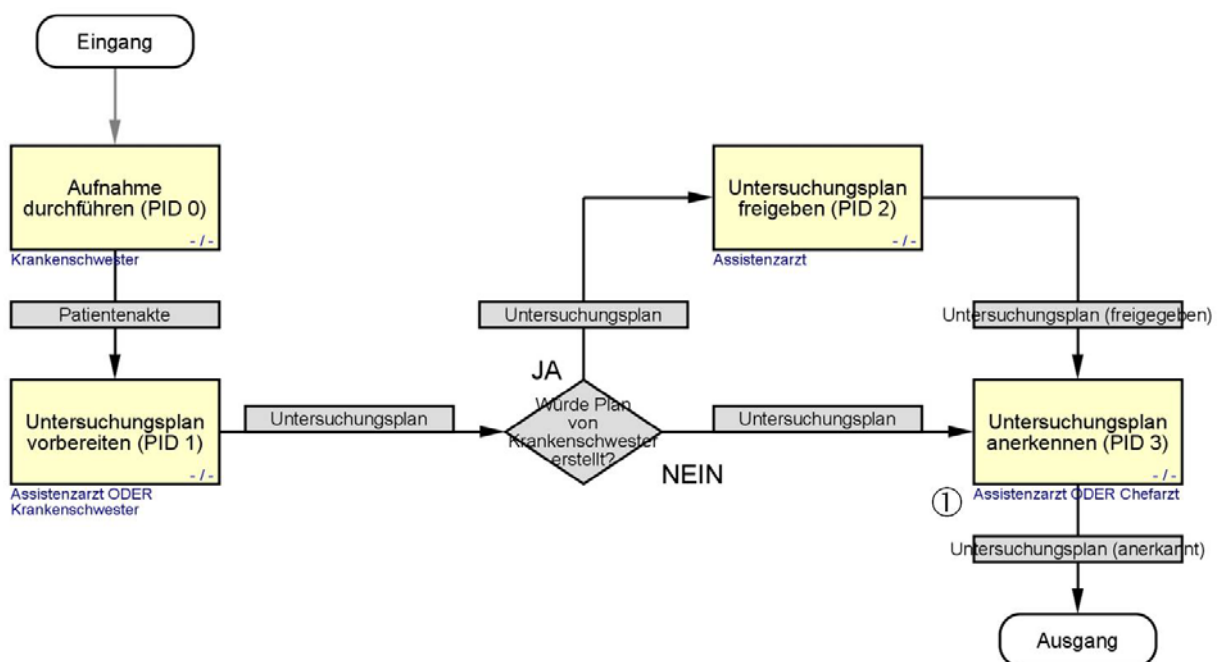


Abbildung 6-1 Entscheidungspunkt mit Alternativen

⁷ Aus implementierungstechnischen Gründen kann bei Verwendung von i>pm die Beschriftung eines Modellierungskonstruktes nicht an deren geometrischen Abmessungen angepasst werden (dies gilt für alle folgenden Prozessabbildungen).

Varianten – variation points und variant options

Als *Variante* wird ein Prozess durch mehrere sog. *variation points* charakterisiert, die in Abbildung 6-2 mit ❶, ❷ und ❸ bzw. ❸* explizit hervorgehoben sind. Die Nummerierungen kennzeichnen die Stellen bzw. Aspekte, in denen sich die in einem integrierten Prozessmodell zusammengeführten Varianten unterscheiden. Jedem dieser variation points sind mindestens zwei oder auch mehrere als *variant options* bezeichnete Auswahlmöglichkeit zugeordnet. Bei variation point ❶ sind dies zum Beispiel Assistenzarzt und Krankenschwester. Jede variant option kann für eine, aber auch für mehrere Varianten als charakterisierende Eigenschaft relevant sein. In dem Beispiel wird jede variant option genau einer der beiden Varianten zugeordnet. Die beiden Ergebnismodelle sind in Abbildung 6-3 und Abbildung 6-4 zu sehen.

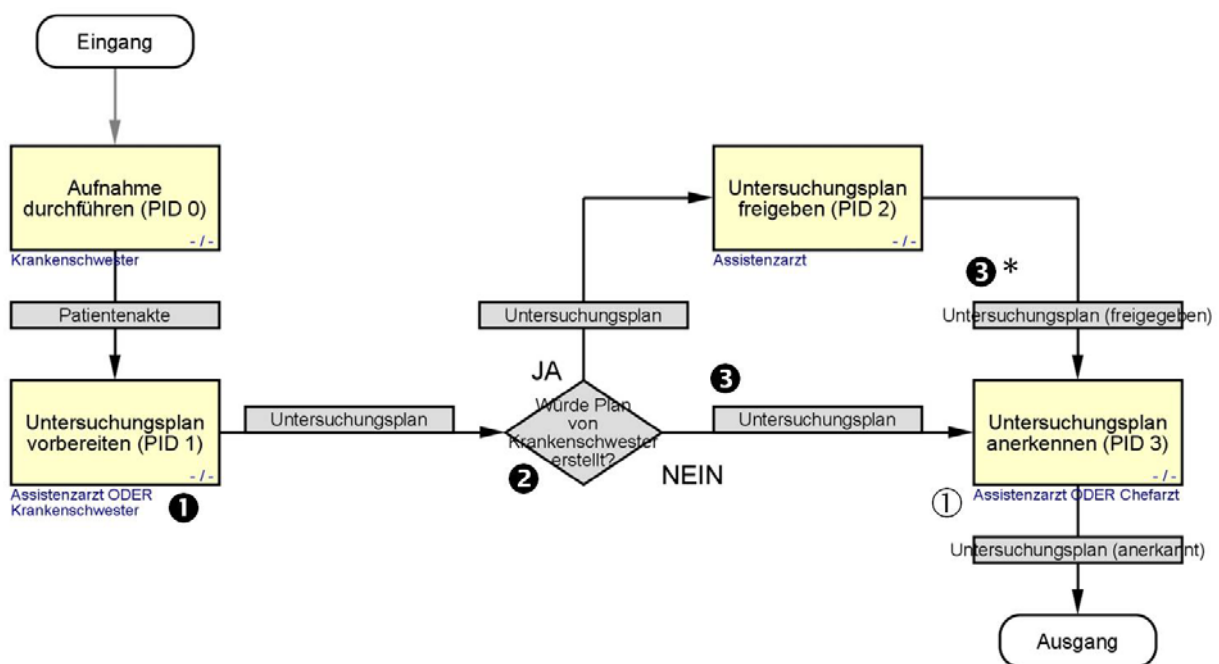


Abbildung 6-2 Varianten mit variation points und variant options

Bei den variation points handelt es sich um Entscheidungen im Bezug auf das *Prozessdesign*. Es werden damit grundlegende Gestaltungsaspekte der Prozesse definiert, denn die Varianten bzw. die dazugehörigen variation points beziehen sich auf die allgemeinen Rahmenbedingungen des Anwendungsfeldes. Die sich aus den Kundenwünschen, Projektvorgaben, Compliance-Anforderungen oder Ähnlichem ergebenden Aspekte lassen sich selten über ein einziges Merkmal inkl. Ausprägung im Prozess abbilden, vielmehr verteilen sie sich auf mehrere dieser variation points.

So eben Gesagtes hat zudem zur Folge, dass die einzelnen variation points auch bei ihrer Auflösung *nicht unabhängig voneinander betrachtet werden*, sondern in ihrer Gesamtheit im Bezug auf eine bestimmte Variante. Die Auswahl einer bestimmten variant option an einem

variation point hat Einfluss auf die Auswahl der variation points an anderen Stellen des Prozessmodells. Damit unterscheiden sie sich wesentlich von den zuvor erläuterten Entscheidungspunkten. Die Auflösung eines variation points kann vor und während der Ausführung eines Prozess erfolgen, je nachdem wann die genaue Ausprägung der variation points definitiv bekannt ist.

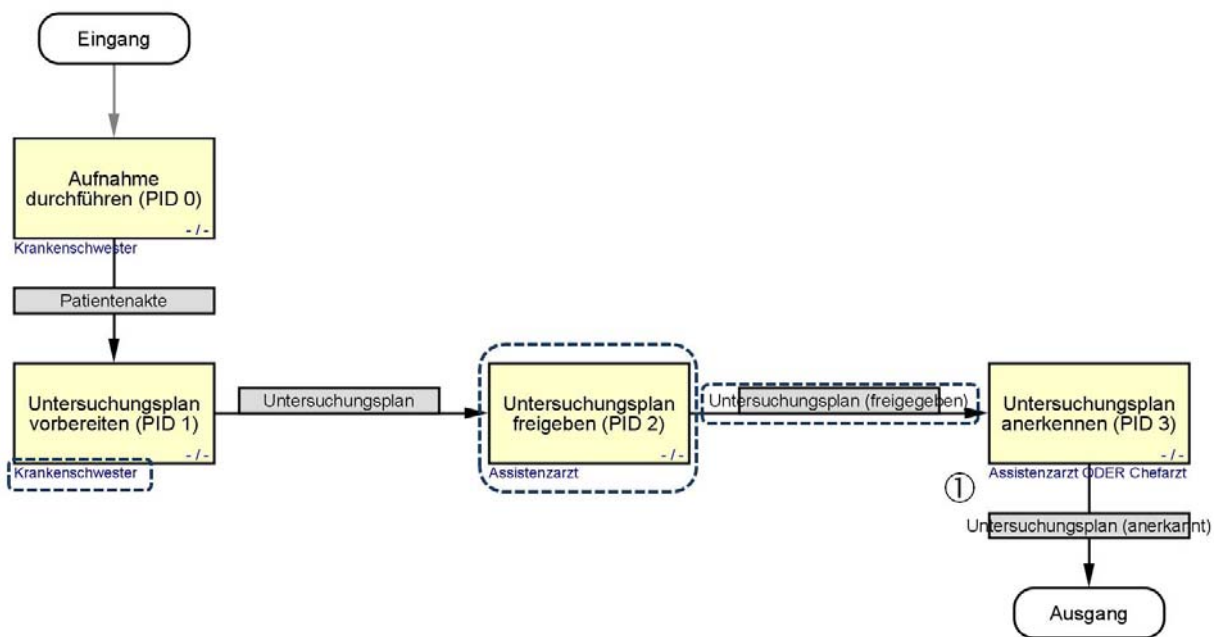


Abbildung 6-3 Variante 1

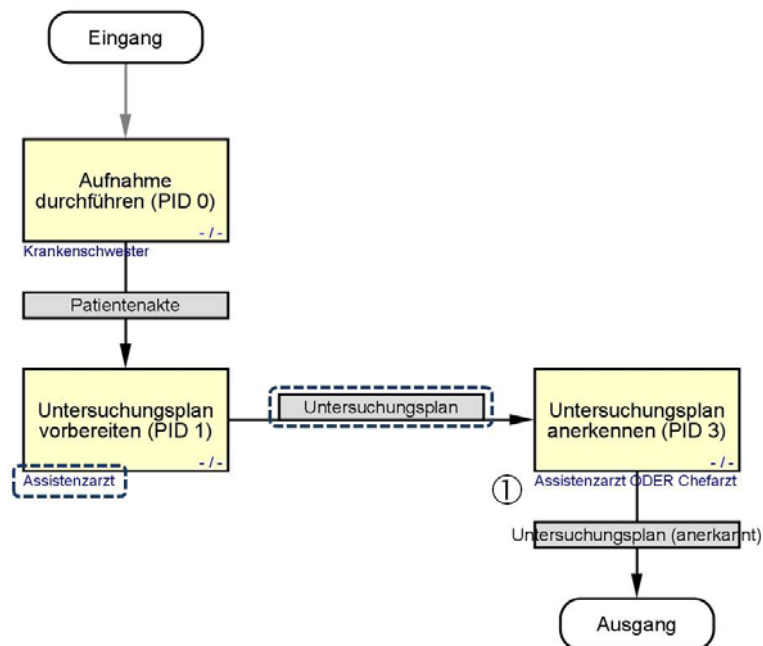


Abbildung 6-4 Variante 2

Jede Variante kann für sich ohne Weiteres auch Entscheidungspunkte enthalten. Dies zeigt sich in dem Beispiel mit dem Entscheidungspunkt ①, der in den beiden einzeln dargestellten Varianten in Abbildung 6-3 bzw. Abbildung 6-4 nicht aufgelöst ist. Wie bereits im vorherigen Abschnitt erläutert, richtet sich die Auflösung dieses Entscheidungspunktes nach dem aktuellen Dienstplan, der erst zum Zeitpunkt der Ausführung eines Prozesses bekannt ist. Diese Entscheidung wird unabhängig davon getroffen, wie der vorherige Prozessablauf dieses Beispiels aussieht. Obwohl es auch den organisatorischen Aspekt betrifft, ist ❶ hingegen als variation point definiert, da es sich um eine variantenrelevante Entscheidung handelt; Variante 1 für die Chirurgie (Abbildung 6-3), Variante 2 für die Innere (Abbildung 6-4). Während in der Chirurgie eine Krankenschwester PID 1 ausführen darf, ist dies bei den Internisten zum Beispiel nicht vorgesehen. Bei der Anerkennung des Untersuchungsplans gehen die beiden Abteilungen in gleicher Art und Weise vor, so dass hier kein Unterscheidungsmerkmal im Sinne eines variation points vorliegt.

6.2.3 Abgrenzung Variante - Version

Im Folgenden wird eine kurze Abgrenzung der Begriffe *Version* und *Variante* erfolgen. In beiden Fällen handelt sich um Objekte, die sowohl identische als auch voneinander abweichende Teile aufweisen. Die dazu notwendigen Änderungen unterscheiden sich jedoch im Hinblick auf den *Faktor Zeit*, d.h. im Hinblick auf den Zeitpunkt, zudem die Änderungen vorgenommen worden sind [Meer05]. Abbildung 6-5 stellt dies schematisch dar.

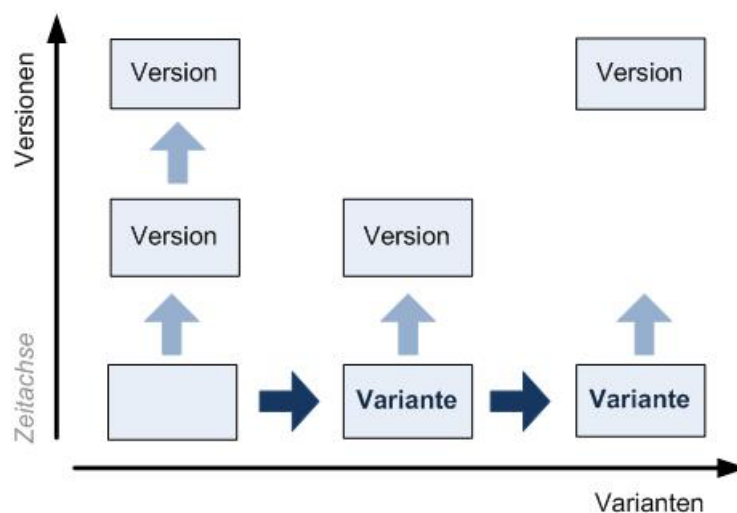


Abbildung 6-5 Versionen – Varianten

Bei einer *Version* werden die Änderungen des Objektes entlang der Zeitachse vorgenommen. Jede Änderung kann zu einem anderen Zeitpunkt erfolgen. Auf einen Zeitraum bezogen gibt

es somit mehrere verschiedene Ausprägungen, sog. Versionen des Objektes. Zu einem bestimmten Zeitpunkt ist zunächst einmal immer eine Version gültig.

Bei den *Varianten* werden die Änderungen an einem Objekt im zeitfreien Sinn vorgenommen [Meer05]. Das geänderte Objekt wird zu einem neuen Objekt, so dass mehrere verschiedene Objekte zeitlich parallel definiert werden [Gabl10] [Meer05]. Varianten ermöglichen somit das bewusste Zulassen mehrerer aktueller Versionen zu einem bestimmten Zeitpunkt. Zu jeder Variante kann zu einem beliebigen Zeitpunkt eine neue Version gebildet werden; es kann zu einer Menge bereits existierender Varianten zu einem späteren Zeitpunkt eine weitere Variante hinzugefügt werden.

Die Konzepte der *Version* und der *Varianten* stehen im übertragenen Sinn orthogonal, d.h. senkrecht zu einander [Wede94], wie es auch in Abbildung 6-5 graphisch dargestellt ist. Im Kontext der Informatik bedeutet dies, dass Versionen und Varianten unabhängig voneinander sind.

6.3 Ansätze zur Modellierung von Varianten

Im Hinblick auf die grundlegende Konzeption variantenreicher Prozessmodelle kann zwischen dem sog. „*Multi-Model*“- und „*Single-Model*“-Ansatz unterschieden werden. Single-Models werden auch als konfigurierbare Prozessmodelle bezeichnet. Beide Ansätze werden hier im Hinblick auf das noch vorzustellende Variantenmanagement-Konzept und das dazu notwendige Datenmodell einander gegenübergestellt.

„*Multi-Model*“-Ansatz

Bei diesem Ansatz wird jede Prozessvariante als eigenes, separates Prozessmodell abgebildet und gespeichert [HaBR10] [WaKü08] [WeRe08]. Diese Vorgehensweise bringt durch die hohe Anzahl an zu erstellenden Modellen einen hohen Modellierungsaufwand mit sich. Es fallen eine große Menge redundanter Daten an, da Varianten per Definition in vielen Teilen übereinstimmen, die jedoch in jedem Modell für eine einzelne Variante wieder erneut modelliert werden müssen. Dabei kann es sehr schnell zu Inkonsistenzen bei eigentlich übereinstimmenden Prozessabschnitten kommen.

Auch die Pflege solcher Modelle ist sehr aufwendig, da variantenübergreifende Änderungen, zum Beispiel aufgrund neuer gesetzlicher Regularien, in alle einzelnen Prozessmodelle ein gepflegt werden müssen. Dabei können sehr schnell Aspekte und/oder Prozesselemente übersehen werden oder Zusammenhänge werden nicht korrekt modelliert.

Durch die separate Abbildung und Speicherung besteht zunächst keine direkte Kopplung zwischen den einzelnen Varianten. Die variantenrelevanten Aspekte bzw. Prozesselemente werden nicht explizit hervorgehoben, was den direkten Vergleich der Varianten erschwert. Zusammenhänge sind damit, nicht nur bei der graphischen Ansicht, schwer zu identifizieren.

Die Varianten können in diesen Fällen unter anderem über Suchanfragen identifiziert werden. Die Suche basiert auf Kriterien, in denen alle gefundenen Prozesse übereinstimmen sollen, im jeweiligen Prozessrepositorium. Einen Einblick kann dazu zum Beispiel [LuSa06a] bzw. [LuSa06b] geben. Von Nachteil ist dabei aktuell, dass sich solche Suchanfragen derzeit meist auf strukturelle bzw. quantitative Aspekte beschränken, während qualitative Aspekte vernachlässigt werden.

„Single-Model“-Ansatz – konfigurierbare Prozessmodelle

Für den „Single-Model“-Ansatz erfolgt die Modellierung der Varianten in einem integrierten (konfigurierbaren) Prozessmodell, wie es in Abbildung 6-2 zu sehen war [ReMT09] [RovA07] [GoAJ08].

Die Stellen, an denen sich die Varianten unterscheiden werden üblicherweise (vgl. zum Beispiel [LaRo08a] oder [PSWW05]) durch sog. variation points explizit hervorgehoben, wie es bereits in Kapitel 6.2 erläutert wurde. Erfolgt dies nicht, weil die Modelle bzw. die Modellierungssprache zum Beispiel keine entsprechende Funktionalität bieten, ist entsprechendes Wissen über den Anwendungsfall essentiell, kann jedoch nicht immer vorausgesetzt werden.

In sehr umfangreichen Prozessmodellen kann der Anwender schnell die Übersicht verlieren, vor allem was die Erkennung einzelner Varianten betrifft. Die Schwierigkeit besteht darin, die richtige Kombination an variant options der verschiedenen variation points zu identifizieren.

Mit der Verwendung konfigurierbarer Modellierungselemente, den variation points, stellt diese Art von Prozessmodellen den Ausgangspunkt zur Ableitung individueller Prozessmodelle, den Varianten, dar. Dies erfolgt durch das Löschen bzw. Deaktivieren nicht-relevanter Prozesselemente, d.h. variant options [Thom05]. Welche Elemente dies genau sind, ist abhängig vom jeweiligen Einsatzzweck bzw. -bereich der resultierenden Variante (zum Beispiel je nach organisatorischer Einheit oder Projekt innerhalb einer Organisation). Zugleich stellt dies einen systematischen Ansatz zur Wiederverwendung von Prozessmodellen dar [LaRo08a] [SaWe97] [TiSo97]. Die Varianten entstehen aus der Kombination von zuvor definierten Prozesselementen.

Konfigurierbare Prozessmodelle werden unter anderem im Kontext von *Referenzmodellen* verwendet [vADR05] [FeLZ05]. Diese werden in Bezug auf eine bestimmte Domäne als ein Repositorium für empfohlene, etablierte Vorgehensweisen angesehen [FeLo03] [RovA07]. Aufgrund ihres generischen Grundkonzeptes ist eine individuelle Anpassung, d.h. Konfiguration, an konkret vorliegende Anforderungen, wie zum Beispiel die eines bestimmten Unternehmens, stets erforderlich [RoSc97]. Als Beispiele für Referenzmodelle, deren Erstellung oftmals durch entsprechende Industriekonsortien vorangetrieben werden, sind das SCOR-Modell [HuSW04] [Step01] für den Bereich des Supply Chain Management,

ITIL [OoGC11] für das Service Management oder das SAP Referenzmodell [Hern05] [CuKe97], das SAP-bezogene Prozesse dokumentiert, zu nennen.

Fazit

In logischer Konsequenz zu den Erläuterungen in Kapitel 6.2.2 wird im Rahmen dieser Arbeit der Single-Model-Ansatz verwendet. Auch hat die Literaturrecherche auf diesem Gebiet gezeigt, dass wesentlich häufiger der „Single-Model“- als der „Multi-Model“-Ansatz verwendet wird (siehe dazu unter anderem [HaBR10], [LaRo08a] oder [GoAJ08]).

Ausschlaggebend ist das Konzept der *Wiederverwendung* [LaRo08a] [ReMA06] [RovA07], durch das bereits vorhandenes Wissen (hier in Form von Prozessen und ihren Aspekten bzw. Prozesselementen) zur Modellierung ähnlicher Prozesse verwendet werden kann [SaWe97] [TiSo97]. Der positive Effekt ist, dass weniger Prozesse bzw. Prozesselemente einzeln definiert werden müssen. Neben der Reduktion des Aufwands führt dies außerdem zu Zeit- sowie Kostenersparnissen. Die Qualität des resultierenden Prozessmodells, der Variante, erhöht sich, da es sich bei einem wiederverwendeten Objekt um ein bereits etabliertes Element handelt, das im Sinne einer Best Practices aufgefasst werden kann [SaWe97] [SmDu01] [ReMA06].

Ein weiterer Faktor, der für den „Single-Model“-Ansatz spricht, ist die Möglichkeit ausgehend von der *integrierten Darstellung durch entsprechende Ableitung zu der isolierten Darstellung einer einzelnen Prozessvariante* zu kommen. Bei der integrierten Darstellung kann ein Gesamtüberblick gegeben werden, bei der isolierten Darstellung ist die Fokussierung auf bestimmte Aspekte bzw. Werte möglich.

6.4 Ansätze des prozessorientierten Variantenmanagements

Das prozessorientierte Variantenmanagement ist in der Wissenschaft wie auch in der Industrie bzw. Wirtschaft ein viel diskutiertes und in zahlreichen (Forschungs-) Projekten bearbeitetes Thema. In diesem Kapitel werden drei Ansätze vorgestellt und bewertet. Sie stammen schwerpunktmäßig aus dem wissenschaftlichen Bereich, sind jedoch in enger Kooperation mit Partnern aus der Industrie entwickelt worden.

6.4.1 C-iEPC

Zu den in Kapitel 3.3.1.2 vorgestellten EPKs wurde im Rahmen diverser Forschungsprojekte von internationalen Universitäten und deren Industriepartnern eine konfigurierbare Version in Form der sog. C-iEPC (engl. Configurable Integrated Event Process Chains) inklusive eines umfassenden Variantenmanagementkonzeptes entwickelt (siehe dazu unter anderem [LaRo08a], [LaRo10], [LLSD07], [vADG10], [LDHM11]). Das Konzept wurde über die EPKs hinaus zudem auf die Workflow-Sprache YAWL (= Yet Another Workflow Language) [vAtH05] angewendet [GoLa08].

Für die Unterstützung des Konfigurationskonzeptes ist inzwischen ein open-Source Werkzeugset (Synergia toolset) von insgesamt sechs zusammenhängenden Werkzeugen vorhanden [LaRo10].

Startpunkt im Bezug auf das Vorgehensmodell ist die Erstellung eines integrierten, konfigurierbaren Prozessmodells unter Verwendung von c-iEPC mit dem *C-iEPC-Designer*. Zur Kennzeichnung der Varianten werden die variablen Prozesselemente (dies können Funktionen, der Kontrollfluss, Konnektoren, Rollen sowie Objekte sein [LDHM11]) in dem Modell explizit als *Variationspunkte* gekennzeichnet. In Abbildung 6-6 sind die konfigurierbaren Prozesselemente an einer dickere Umrandung zu erkennen⁸. Jedem konfigurierbaren Prozesselement sind mindestens zwei Konfigurationsmöglichkeiten zugeordnet.

Das resultierende Modell, das auch als Referenzmodell bezeichnet wird, kann damit gemäß den individuellen Bedürfnissen des Anwenders konfiguriert werden (*Process Configurator*)⁹ und es erfolgt letztendlich die Definition einer konkreten Prozessvariante (*Process Individualizer*). Je nachdem ob zum Beispiel eine Funktion auf ON oder OFF gesetzt wird, erfolgt eine Anzeige im finalen Prozessmodell oder nicht. Mit OPT kann diese Entscheidung auf den Zeitpunkt der Ausführung verschoben werden [LaRo08a]. Bei einem konfigurierbaren XOR-Konnektor kann entweder der rechte oder der linke ausgehende Zweig entfernt werden oder der konfigurierbare Konnektor wird zu einem nicht-konfigurierbaren Konnektor umfunktioniert, so dass auch hier die Entscheidung erst zur Ausführungszeit getroffen wird. Um nicht auf alle verschiedenen konfigurierbaren Prozesselemente und ihre speziellen Eigenschaften eingehen zu müssen, sei an dieser Stelle auf [LDHM11] verwiesen.

Eine *gestufte Konfiguration* kann insofern realisiert werden als in der Zeit vor der Prozessausführung nicht alle konfigurierbaren Elemente auf einmal aufgelöst werden müssen, sondern zunächst nur ein Teil. Funktionen, die erst zur Ausführungszeit aufgelöst werden, sind daher extra mit „OPT“ gekennzeichnet. Bei jedem einzelnen Konfigurationsschritt wird dabei auf der Basis der Angabe entsprechender Bedingungen (process constraints) automatisch die syntaktische wie semantische Korrektheit der Modelle überprüft [vADG10]. Ein Modell ist zum Beispiel unter anderem nur dann syntaktisch korrekt, wenn jede Funktion mindestens eine ein- und eine ausgehende Verbindungskante besitzt.

Für den Konfigurationsvorgang werden im Hinblick auf den Anwender, der nicht unbedingt Modellierungsexperte sein muss, die C-iEPCs um einen *fragebogenbasierten Ansatz* ergänzt [LaGD07] [LLSD07]. Die Fragen bzw. die jeweils möglichen Antworten des interaktiven Fragebogens sind direkt mit den Variationspunkten des Prozessmodells und deren möglichen

⁸ Die einzelnen Bezeichnungen der Prozesselemente sind in diesem Kontext nicht relevant.

⁹ Dieser *process configurator* darf nicht mit dem in dieser Arbeit in 6.5 vorgestellten Prozesskonfigurator verwechselt werden. Bei Ersterem handelt es sich um eine Teilkomponente, bei Letzterem um ein Gesamtkonzept.

Ausprägungen verknüpft. Dabei werden auch Abhängigkeiten zwischen verschiedenen Auswahlmöglichkeiten bzw. Ausprägungen der einzelnen Variationspunkte berücksichtigt. Diese haben Einfluss auf die Reihenfolge der gestellten Fragen. In dem Konfigurationswerkzeug sind dazu eine interaktive Komponente für die Anwendung der Fragebögen (*Quaestio*) und deren initiale Erstellung (*Questionnaire Designer*) vorhanden [LDHM11].

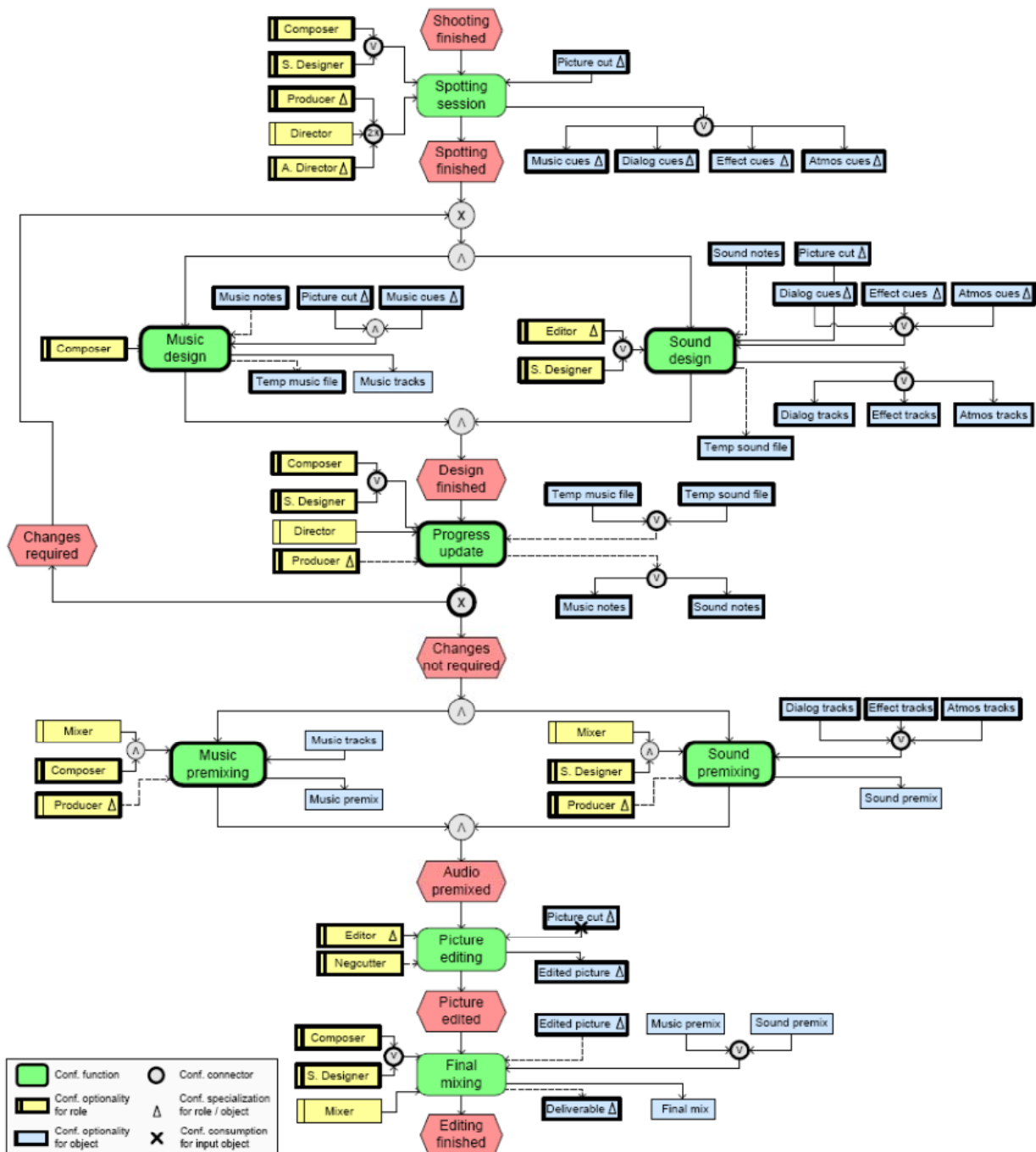


Abbildung 6-6 C-IEPC aus [LDHM11]

Bewertung

EPKs bzw. ihre Erweiterung als c-iEPC als Basis eines Prozesskonfigurationsansatzes zu verwenden bietet den Vorteil, dass es sich um eine Modellierungssprache handelt, die in der Praxis sehr häufig zur konzeptionellen Modellierung im Rahmen der Entwicklung von Informationssystemen eingesetzt wird. Informationssysteme wiederum dienen letztendlich der Unterstützung der Prozessausführung.

Da die Konfigurierbarkeit nicht nur, wie sonst oftmals üblich, bzgl. des funktionalen Aspekts definiert ist, sondern auch in Bezug auf weitere Aspekte, wie zum Beispiel Rollen und Objekte, handelt es sich im Bezug auf die üblicherweise bei Prozessen relevanten Perspektiven [JaBu96] um einen sehr umfassenden Ansatz.

Dass es sich nicht um einen nur auf EPCs anwendbaren Ansatzes handelt, konnte durch die Anwendung auf die Workflow-Sprache YAWL gezeigt werden (siehe [GoAJ08] und [GoLa08] für detaillierte Informationen).

Mittels des dazugehörigen Fragebogens für die Durchführung der Konfiguration kann von der eigentlichen Modellierungssprache, die vor allem für Anwender ohne technischen Hintergrund schwer zu verstehenden ist, abstrahiert werden. Für die Beantwortung der Fragen ist einzig und allein domänenspezifisches Wissen notwendig bzw. ausreichend. Die Variabilität der Prozessmodelle inklusive der Abhängigkeiten zwischen den Auswahlmöglichkeiten unterschiedlicher Variationspunkte kann außerhalb des eigentlichen Prozessmodells festgehalten werden, was dessen Verwendung wesentlich erleichtert.

Desweiteren wird der Anwender mit Hilfe des Fragebogens durch den Konfigurationsprozess geleitet. Es kann damit sicher gestellt werden, dass die konfigurierten Prozessmodelle später den Geschäftsbedingungen bzw. -regeln der Anwendungsdomäne entsprechen. Von Vorteil ist außerdem, dass der fragebogenbasierte Ansatz so allgemeingültig definiert ist, dass er sich auch auf andere Modelltypen, wie zum Beispiel Datenmodelle im Allgemeinen anwenden lässt.

Die Konfiguration wird prinzipiell als Designaufgabe angesehen. Deren komplette Ausführung kann sich dabei bis in die Ausführungszeit hinein erstrecken. Aufgrund der umfangreichen Werkzeugunterstützung handelt es sich um einen sehr umfassenden Ansatz. Zur Abrundung des Konzeptes tragen auch die drei im Rahmen dieses Gesamtprojektes durchgeführten Studien zur Überprüfung der erarbeiteten Konzepte bei.

Der Ansatz wird in seiner Flexibilität dadurch eingeschränkt, dass vor dem Beginn der Ausführung festzulegen ist, ob variable Stellen erst zur Ausführungszeit aufgelöst werden können. Dies schränkt die Handlungsfreiheit je nach aktuellem Informationsstand ein. Zudem ist für die Darstellung des Modellinhaltes nur eine einzige Darstellungsform möglich, die alle notwendigen Informationen integrieren muss. Dadurch sind die Modelle mit Informationen überladen und die Verständlichkeit leidet.

6.4.2 PESOA

Auch im Bereich des Softwareengineerings werden Varianten explizit behandelt. Ein Konzept zur Modellierung von Prozessvarianten in UML- oder BPMN-Modellen stellt PESOA (= Process Family Engineering in Service Oriented Applications) dar [PSWW05] [ScPu06].

Ziel des Ansatzes bzw. Projektes ist es nicht, eine spezielle (Prozess-) Modellierungssprache zur Generierung und Konfiguration variantenreicher Prozessmodelle zu liefern. Der Fokus liegt auf der Architektur prozessorientierter Softwaresysteme und deren notwendigen, kontextbedingten Anpassungen, so dass Varianten entstehen. Diese Softwaresysteme werden dabei jedoch auf der Basis von Prozessmodellen erstellt, weshalb im Rahmen der vorliegenden Arbeit nur auf den Aspekt eingegangen werden soll, wie die Prozessvariabilität in den Modellen festgehalten werden kann, während die weiteren Aspekte der Softwaresysteme nicht erläutert werden.

Die mit BPMN oder UML (Aktivitätsdiagramme) erstellten Prozessmodelle werden bei diesem Ansatz durch *Annotationen* erweitert. Die variablen bzw. zu konfigurierenden Stellen im Prozess werden als *variation points* bezeichnet und mit dem Stereotypen <<VarPoint>> gekennzeichnet [PSWW05], wie es in Abbildung 6-7 zu sehen ist. Die zu annotierenden Prozesselemente sind in einem BPMN-Modell die Tasks, in UML-Aktivitäts-Diagrammen die Aktivitäten. Mit dieser expliziten Kennzeichnung werden die verschiedenen Konfigurationsoptionen spezifiziert [ScPu06] und diverse Variabilitätsmechanismen, wie zum Beispiel Vererbung oder Parametrisierung, in die Prozessmodelle integriert [PSWW05].

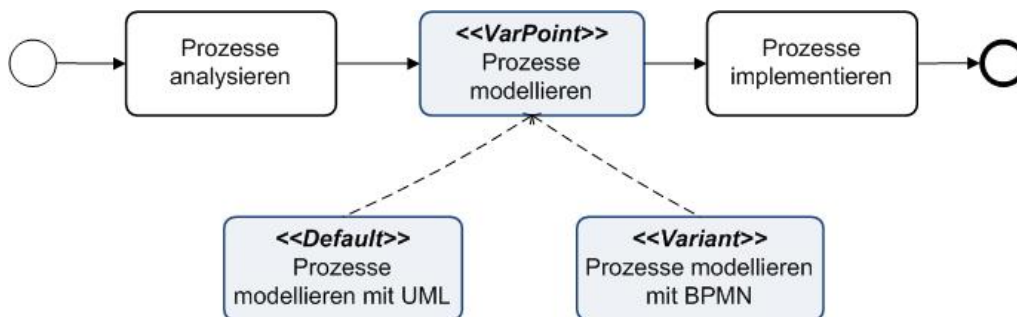


Abbildung 6-7 Nach PESOA annotiertes BPMN-Prozessmodell

Die an den variation points vorhandenen Auswahlmöglichkeiten werden diesen jeweils als <<Variant>> zugeordnet, wie es in dem Ausschnitt aus einem BPMN-Prozessmodell in Abbildung 6-7 zu sehen ist. Auswahlmöglichkeiten, die den Normalfall darstellen bzw. üblicherweise ausgewählt werden, können mit <<Default>> entsprechend markiert werden. Handelt es sich um eine exklusive Variante, wird statt <<VarPoint>> der Stereotyp <<Abstract>> oder <<Alternative>> verwendet. Optionale Varianten werden mit <<Null>> gekennzeichnet. Mit der Durchführung der Konfiguration wird jeder variation point

typgerecht angepasst, so dass ein individualisiertes Prozessmodell, d.h. eine Variante, entsteht. Zur Werkzeugunterstützung ist ein Eclipse Plugin vorhanden.

Bewertung

Aufgrund der verschiedenen möglichen Variabilitätsmechanismen (Vererbung, Parametrisierung, ...) werden die Varianten nicht einheitlich beschrieben, was sich negativ auf die Verständlichkeit und die Einfachheit des Ansatzes auswirkt. Basisprozess und die eigentlichen Varianten können weder logisch noch modellierungstechnisch voneinander getrennt werden, wodurch die Darstellungen weniger übersichtlich sind. Verschiedene Perspektiven, wie sie bei Prozessen üblicherweise relevant sind [JaBu96], werden hier nicht berücksichtigt. Dies ergibt sich jedoch daraus, dass nicht die Prozesse direkt, sondern vielmehr prozessorientierte Softwaresysteme betrachtet werden, für welche die perspektivenorientierte Betrachtung weniger relevant ist.

Der Anwender wird bei der Durchführung der Konfiguration in keiner Weise unterstützt bzw. geleitet, wie es zum Beispiel bei dem Konfigurationskonzept zu den C-iEPCs (siehe Kapitel 6.4.1) der Fall ist. Das Konfigurieren beschränkt sich auf das Entfernen von Prozesselementen; ein Verschieben oder Ändern ist nicht möglich. Zudem ist das Entfernen nur bzgl. Tasks (BPMN) oder Aktivitäten (UML) möglich; andere Prozesselemente können nicht konfiguriert werden.

6.4.3 Provop

Einen weiteren Variantenmanagementansatz stellt der sog. Provop-Ansatz dar [HaBR08d] [HaBR10]. Hier steht die Konfiguration von Varianten mittels eines operationalen Ansatzes durch Anwendung von kontextbezogenen Änderungsoperationen an einem variantenreichen Prozessmodell im Vordergrund. Ziel des Ansatzes ist es, über den Aspekt der Modellierung hinaus den gesamten Prozesslebenszyklus im Bezug auf das Thema der Varianten zu unterstützen.

Als Ausgangspunkt wird ein Prozessmodell verwendet, das alle Varianten integriert. In diesem sog. Basisprozess werden sog. *Aufsetzpunkte* (engl. *adjustment points*) zur Darstellung der konfigurierbaren Stellen definiert [HaBR08b] [HaBR08d] [HaBR10]. Sie können an Aktivitäten sowie Verzweigungen modelliert werden. In Abbildung 6-8 sind sie als schwarze Rauten inklusive der Benennung in alphabetischer Reihenfolge zu sehen.

An diesen Aufsetzpunkten werden die für die Konfiguration notwendigen *Anpassungen* vorgenommen. Es besteht die Möglichkeit Prozesselemente (Aktivitäten oder Verzweigungen) zu löschen, ein zusätzliches Element einzuführen oder existierende Elemente zu verschieben. Außerdem können die Attribute der Prozesselemente angepasst werden [HaBR08b] [HaBR08c] [HaBR08d] [HaBR10]. Dabei wird explizit definiert, unter welchen

Kontextbedingungen die Anpassungen erfolgen sollen. Die Kontextbedingungen werden mittels Variablen und dazugehörigen Wertebereichen beschrieben [HaBR08c].

Zur besseren Strukturierung werden die Anpassungen an einem Aufsetzpunkt jeweils zu sog. *Optionen* zusammengefasst [HaBR08b] [HaBR08d] [HaBR10], die eigenständige Repräsentations-Objekte darstellen. Sie werden vollständig, teilweise minimalisiert oder komplett minimalisiert im Prozessmodell angezeigt. Eine vollständige Darstellung ist in Abbildung 6-8 zu sehen.

Die Anwendung ein oder mehrerer aufeinanderfolgender Optionen führt schließlich zu einer konkreten Variante [HaBR08c] [HaBR10]. Dazu gilt es, über Attribute die statischen Kontextbedingungen, die sich während der Ausführung der Prozesse nicht mehr ändern, konkret festzulegen. In beiden Fällen muss darauf geachtet werden, dass dabei keine Widersprüche in der Logik der Prozessmodelle entstehen. Wie bei den Optionen an sich gibt es auch bei den entstehenden Ergebnismodellen verschiedene Möglichkeiten der Darstellung; unter anderem ist dies mit und ohne Hervorhebung der angewendeten Optionen möglich [HaBR08d].

Für die Modellierung und die anschließende Konfiguration ist ein Werkzeug-Prototyp vorhanden, der auf dem *ARIS Business Architect* aufbaut [HaBR10]. Mit diesem können BPMN-Prozessmodelle gemäß dem definierten Konzept erstellt und konfiguriert werden.

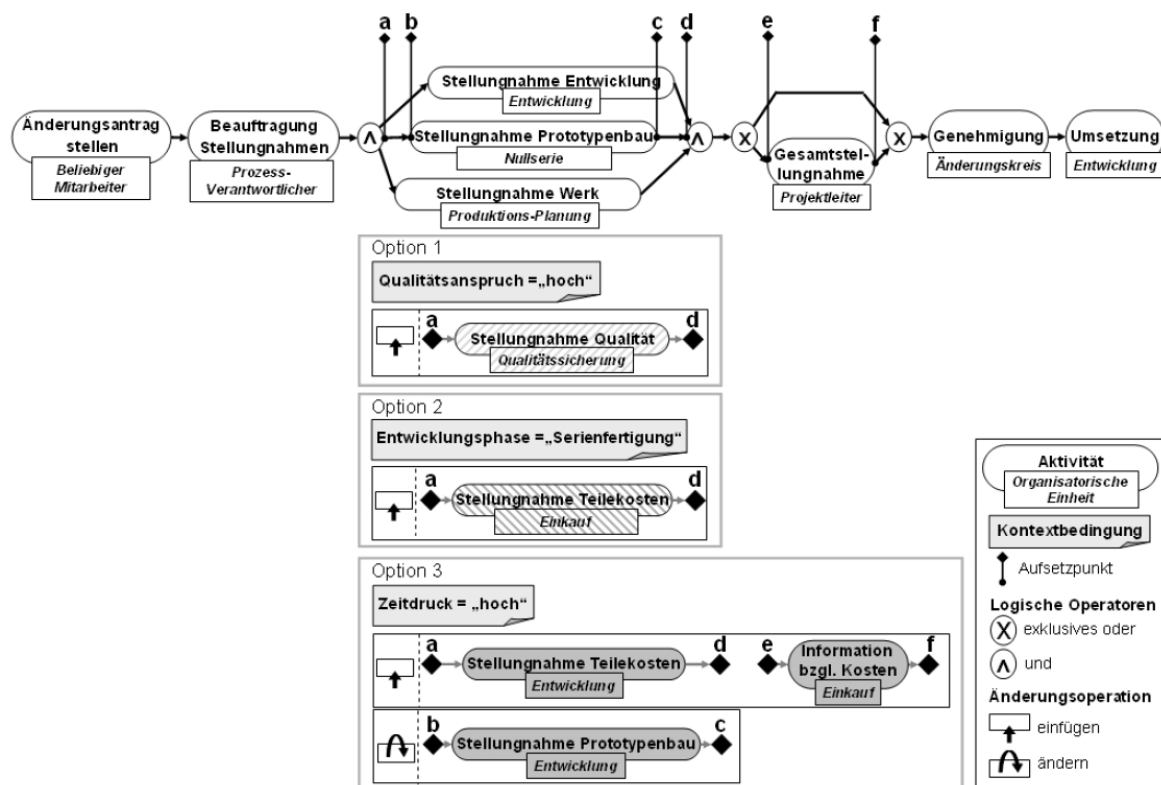


Abbildung 6-8 Beispiel mit Aufsetzpunkten aus [HaBR08a]

Während der *Ausführungszeit* werden die als dynamisch bezeichneten Kontextvariablen relevant, deren Werte sich zur Ausführungszeit ändern können [HaBR08c]. Durch deren Anpassungen ist ein Wechsel zwischen den Varianten zur Ausführungszeit möglich, was einer Re-Konfiguration der Varianten „on-the-fly“ entspricht. Damit soll die Unterstützung in Bezug auf Varianten entlang des gesamten Prozesslebenszyklus ermöglicht werden. Diese Ideen sind bisher nur konzeptionell ausgearbeitet. Eine entsprechende Tool-Realisierung ist bisher nicht vorhanden.

Bewertung

Mit der angestrebten Unterstützung des Variantenmanagements über den gesamten Prozesslebenszyklus, handelt es sich in dieser Hinsicht um einen sehr vollständigen Ansatz. Er bietet dem Anwender ein großes Maß an Handlungsfreiheit, da er sich bei dem Start der Ausführung noch nicht auf genau eine Variante festgelegt haben muss; er kann die Entscheidung kurzfristig vornehmen, nämlich dann wenn die notwendigen Informationen vorhanden sind.

Durch die Möglichkeit zur Anpassung der Attribute eines Prozesses wird mehr als nur der funktionale Aspekt berücksichtigt. Da jedoch Daten, Rollen oder Werkzeuge nicht über die Attribute der Prozesse angegeben werden und diese Perspektiven auch nicht anderweitig bei diesem Ansatz berücksichtigt werden, ist damit die Mächtigkeit des Ansatzes im Bezug auf die Unterstützung von Prozessperspektiven gering.

Die Optionen werden als separate Objekte zu dem eigentlichen Prozessmodell definiert. Die genaue Zuordnung erfolgt über die in den Prozessmodellen explizit modellierten Aufsetzpunkte. Damit wird mit den Optionen die Variabilität der Prozesse außerhalb des Modells verlagert, wodurch die Modelle an sich zwar gut lesbar bleiben, die Einfachheit sowie Verständlichkeit des Ansatzes im Gesamten jedoch leidet, da der Zusammenhang zwischen Optionen und Prozessmodell erst einmal verstanden werden muss. Die separate Anzeige der Optionen in dem Prozessmodell ist dennoch notwendig. Dies nimmt zusätzlichen Platz in Anspruch und verringert die Übersichtlichkeit der graphischen Darstellung, was für Anwender mit wenig bis gar keiner Modellierungserfahrung eine Schwierigkeit darstellen kann. Die kombinierte Anwendung von Optionen birgt zudem Herausforderungen hinsichtlich der Verständlichkeit. Zudem muss überprüft werden, ob keine redundanten, widersprüchlichen und kommutativen Anpassungen vorhanden sind.

Der Ansatz ist unabhängig von einer konkreten Prozessmodellierungssprache entwickelt worden. Bei der Anwendung im Rahmen eines Prototyps mussten die EPKs als beispielhaft anzuwendende Modellierungssprache bereits aufgrund fehlender Eigenschaften bzgl. des Variantenmanagement-Ansatzes ausgeschlossen werden [HaBR10], so dass dieser Ansatz im Bezug auf Modellierungssprachen nicht ausreichend generisch definiert ist.

6.4.4 Fazit

Die Bewertung der existierenden Ansätze zur Erstellung konfigurierbarer Prozessmodelle zeigt einige Schwachstellen auf. So stellt zum einen die verständliche und übersichtliche, kompakte und zugleich ausdrucksstarke *Darstellungsform* der Varianten und ihrer *variation points* inklusive der vorhandenen *Abhängigkeiten* eine Herausforderung dar. Zudem sollte die Variabilität bzgl. jeder beliebigen *Perspektive* definiert werden können.

Bei allen drei Ansätzen sind Möglichkeiten zur Anzeige der *variation points* definiert. Jedoch fehlt es bei den C-iEPC an der Möglichkeit, die Abhängigkeiten zwischen den *variation points* direkt in dem Modell mit abzubilden. Das Modell ist zudem durch die zusätzliche Anzeige der variablen Stellen sehr mit Informationen überladen, so dass die Übersichtlichkeit leidet. Provop zeigt die Konfigurationsmöglichkeiten inklusive der Abhängigkeiten außerhalb des Modells an, was die Einfachheit und Verständlichkeit des Ansatzes bei dessen Verwendung reduziert. Zudem können Perspektiven von Prozessen bisher nur eingeschränkt abgebildet werden.

Aufgrund der meist hohen Anzahl an *variation points* und den dazugehörigen Auswahlmöglichkeiten inklusive Abhängigkeiten zwischen diesen, stellt die *Konfiguration* der Prozessmodelle als Entscheidungsprozess für den Anwender ebenfalls eine Herausforderung dar. Es sollte sichergestellt sein, dass ein Anwender nur dann eine variantenrelevante Entscheidung trifft, wenn er die dazu *notigen Informationen* hat, was möglicherweise erst in der Phase der Prozessausführung der Fall ist.

C-iEPC hat zwar einen gestuften Konfigurationsprozess definiert, doch muss vor dem Start der Konfiguration bereits festgelegt werden, welche *variation points* vor und welche erst während der Ausführung aufgelöst werden müssen. Dazu muss bekannt sein, wann die Informationen jeweils zur Verfügung stehen, was nicht immer mit Sicherheit vorhersehbar ist. Bei Provop muss vor dem Start der Ausführung eine Variante definiert sein. Ein Wechsel zwischen Varianten während der Ausführungszeit ist zwar prinzipiell vorgesehen, doch dann mit viel Aufwand verbunden, weil einzelne Prozessabschnitte gegebenenfalls nochmals durchlaufen werden müssen.

Die Ansätze weisen bzgl. Darstellung in einem Modell und dem Ableiten von Varianten aus dem selbigen jeweils noch Lücken auf. Diese gilt es mit dem nun folgenden Ansatz zu schließen.

6.5 Der Prozesskonfigurator

Mit dem nun vorzustellenden *Prozesskonfigurator* sollen die Anwender methodisch sowie werkzeugtechnisch beim Management von Prozessvarianten unterstützt werden.

Da bisher nur individuelle Lösungen zur Modellierung von Prozessvarianten existieren, denen es zudem an Kompaktheit, Ausdrucksstärke und/oder Benutzerfreundlichkeit mangelt, wurde

ein *generisches, konfigurierbares Datenmodell* konzipiert. Selbiges lässt sich auf jede beliebige (Prozess-) Modellierungssprache anwenden.

Für eine umfassende Unterstützung des Anwenders reicht dies jedoch nicht aus, da die schwierige Aufgabe der Konfiguration noch bewältigt werden muss. Dabei gilt es den scheinbaren Widerspruch zwischen der Führung durch den Entscheidungsprozess und der Gewährung maximaler Handlungsfreiheit aufzulösen. Gerade letzteres wird bei den existierenden Lösungen vernachlässigt. Die methodische Unterstützung erfolgt in Form eines *gestuften Konfigurationsprozesses*.

Zunächst werden die Anforderungen an den Prozesskonfigurator erläutert. Ziel war es, die Anforderungen domänenübergreifend zu identifizieren. Damit sollte ein möglichst generisches Konzept entwickelt werden, das sich beispielsweise auf die in Kapitel 4 vorgestellten (Modellierungs-) Projekte anwenden lässt. Bei der anschließenden Erläuterung des Konzeptes wird vor allem auf den gestuften Konfigurationsprozess und das konfigurierbare Datenmodell eingegangen. In dem Zusammenhang wird auch der erste prototypische Ansatz der Integration in ein Modellierungswerkzeug vorgestellt.

6.5.1 Anforderungen an den Prozesskonfigurator

Zur besseren Übersichtlichkeit wurde die Menge der identifizierten Anforderungen in die Kategorien *Grundprinzip*, *Funktionalität*, *Architektur* und *Datenbasis* unterteilt (Abbildung 6-9). Die einzelnen Kategorien werden im Folgenden genauer erläutert.

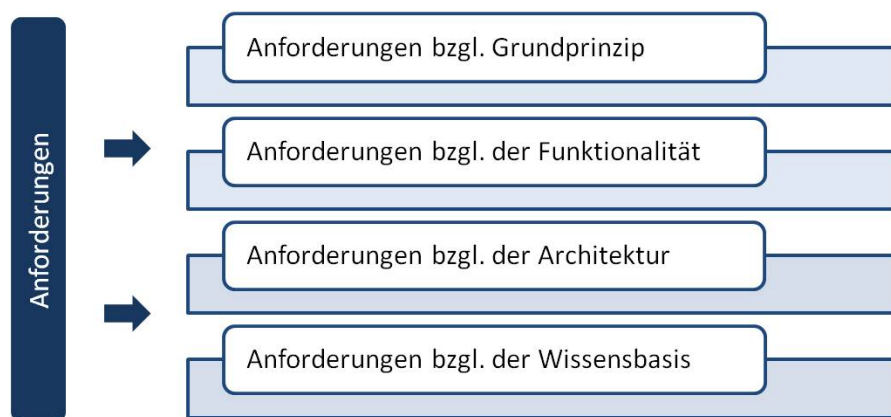


Abbildung 6-9 Kategorisierung der Anforderungen

Grundprinzip

- Der vorgesehene Anwendungsbereich impliziert, dass es sich um einen *prozessorientierten* Konfigurator handeln muss, der die Charakteristika von Prozessen berücksichtigt.

- Für ein umfassendes Variantenmanagement reicht es nicht aus, nur die funktionale Perspektive der Prozesse zu berücksichtigen. Vielmehr müssen auch der datenorientierte, der organisatorische, operationale und sonstige Aspekte beachtet werden (*Multi-Perspektivenorientiert*) [BoRB05] [CuKO92] [Dave05] [JaBu96] [LDHM11] [Sche00].
- Da in den verschiedensten Domänen ein prozessorientiertes Variantenmanagement notwendig ist [MuRB08], sollte der Konfigurationsansatz *domänenunabhängig* sein [SaWe97], d.h. zum Beispiel in der Medizin, in der Verwaltung wie in der Produktentwicklung einsetzbar sein. Zugleich muss dennoch eine *flexible Anpassung* an domänen- oder auch unternehmensspezifische Besonderheiten möglich sein, da sich die Varianten in ihren konkreten Aspekten sehr unterscheiden können.
- Der Einsatz des Prozesskonfigurators sollte für Prozessmodelle möglich sein, die mit einer *beliebigen Prozessmodellierungssprache* generiert worden sind.
- Der Prozesskonfigurator sollte den gesamten *Prozesslebenszyklus* (siehe Kapitel 2.3) unterstützen. Zum einen soll sich die Möglichkeit zur Konfiguration nicht nur auf den *Zeitpunkt der Instanziierung* der Prozesse beschränken, sondern weitere Konfigurationsschritte auch noch zum *Zeitpunkt der Ausführung* erlauben. Erst dann hat der Anwender in manchen Fällen ausreichend Informationen. Zum anderen wird mit der Berücksichtigung des Prozesslebenszyklus das *Re-Design* der Prozesse bzw. Varianten mit eingeschlossen, das auf Basis der Informationen aus dem Monitoring und Controlling erfolgt. Dies ist zum einen im Hinblick auf Verbesserungen bzw. Änderungen notwendig, die im Laufe der Zeit an einem Prozessmodell vorgenommen werden müssen; zum anderen bietet es dem Anwender die Möglichkeit nach einem Zyklus zu einer anderen Variante zu wechseln.

Funktionalität

- Die *variation points* und die relevanten *variant options* müssen den Nutzern des Prozesskonfigurators explizit und anschaulich präsentiert werden.
- Für eine leichtere und zielorientiertere Auswahl seiner Varianten mit ihrer besonderen Charakteristik sollte der Konfigurationsprozess strukturiert, d.h. in einzelne Abschnitte unterteilt werden können. Damit kann sich der Anwender der Lösung schrittweise nähern, und es ist möglich den Anwender durch den Konfigurationsprozess zu leiten (*gestufte Konfiguration*) [CzHE04] [CzHE05].
- Mit Letzterem geht einher, dass der Anwender auch über die *Konsequenzen einer Entscheidung* (bzgl. einer Stufe innerhalb des Konfigurationsprozess) informiert werden soll. Dies ist notwendig aufgrund von Abhängigkeiten zwischen den *variation points*.
- Welche *variation points* auf einer Stufe aufgelöst werden, soll der Anwender *individuell bestimmen* können. Die *Reihenfolge* der Auflösung soll also nicht durch

den Kontrollfluss der Prozesse vorgegeben werden. Damit soll es möglich sein, die Charakteristiken solcher Varianten, die eine hohe *Priorität* aufweisen, so früh wie möglich zu fixieren, ohne das vorherige Konfigurationsentscheidungen dies verhindern. Damit soll sichergestellt werden, dass die finale Variante die für den Anwender wichtigen Eigenschaften sicher aufweist.

- Es sollte ein *interaktiver* Konfigurationsprozess sein, der Prozessmodellierer, Domänenexperten und den eigentlichen Prozessausführenden mit einbezieht.
- Die Ergebnisse der einzelnen Konfigurationsstufen müssen dargestellt werden können.
- Eng verbunden mit dem interaktiven Konfigurationsprozess ist die Notwendigkeit einer *Validierung* der Prozessmodelle, um die Korrektheit sowie Vollständigkeit der Prozesse garantieren zu können [LaVi01] [LiSS94].

Architektur

- Es muss die Möglichkeit bestehen, den Prozesskonfigurator an weitere (funktionale) Anforderungen, zum Beispiel durch das Hinzufügen von speziellen Applikationen, anpassen zu können. Dazu ist eine *modulare Architektur* von Vorteil.
- Eine *Datenbasis* inklusive des Konzeptes eines Datenmodells zur Speicherung aller Prozessmodelle und deren Varianten ist notwendig.
- Eine *Konfigurationskomponente* ist notwendig.
- Eine *Visualisierungskomponente* zur Darstellung der Prozessmodelle zum Zweck der Ergebnispräsentation oder bei der Modellierung ist notwendig; ebenso ist dies zur Durchführung des Konfigurationsprozesses erforderlich.

Datenbasis

- Die Datenbasis muss *alle Prozesse und deren Elemente*, deren Varianten inklusive der einzelnen variation points enthalten.
- Die Datenbasis soll *unabhängig* von einer Prozessmodellierungssprache sein.
- Die Datenbasis muss *konfigurierbar* sein.
- Die Konzeptionalisierung der zu konfigurierenden Prozesselemente soll auf Basis eines *modellbasierten Ansatzes* erfolgen. Dies ermöglicht die Komposition verschiedenster Komponenten unter Berücksichtigung der Beziehungen, die diese untereinander haben. Die gewünschten Einzelobjekte, hier die einzelnen Prozessvarianten, können daraus dann entsprechend, im Sinne einer Spezialisierung, abgeleitet werden [Bode06] [MiFr89] [SaWe97].
- Der modellbasierte Ansatz wird außerdem durch den Aspekt der *Wiederverwendung* motiviert; jede konfigurierte Variante wird dann als eine individuelle Kombination (= Komposition) der zuvor in der Datenbasis definierten Elemente definiert [BDDK04] [SaWe97] [TiSo97].

Im Folgenden werden aus ressourcentechnischen Gründen nicht alle Punkte umfassend behandelt. Der Schwerpunkt liegt auf der Ausarbeitung der *Datenbasis* sowie auf Teilaspekten der *Funktionalität*. Zu den entwickelten Konzepten ist bereits ein erster Prototyp zur Modellerstellung vorhanden, während die Toolunterstützung für den Konfigurationsvorgang noch fehlt.

6.5.2 Das Konzept des Prozesskonfigurators

Abbildung 6-10 gibt einen Überblick über das Konzept des Prozesskonfigurators inklusive der methodischen Vorgehensweise zu dessen Verwendung. Hier werden hauptsächlich die Anforderungen bzgl. der Funktionalität aus Kapitel 6.5.1 umgesetzt. Auf die Datenbasis wird ab Kapitel 6.5.3. eingegangen.

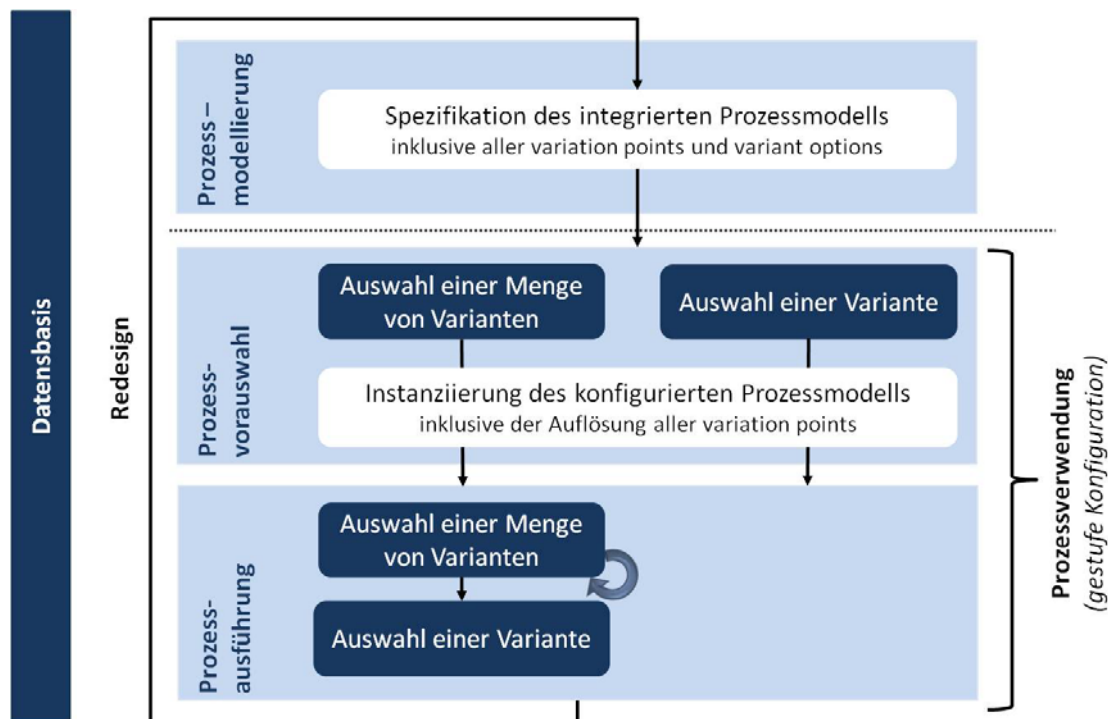


Abbildung 6-10 Konzept des Prozesskonfigurators

6.5.2.1 Prozessmodellierung

Das gesamte Vorgehen beginnt zum Zeitpunkt der Modellierung. Das Ziel ist, ein *variant processmodel vPM*, das alle Varianten gemäß dem ‚Single-Model‘-Ansatz (Kapitel 6.3) in ein Modell integriert, zu definieren. Abbildung 6-11 bildet den nun zu erläuternden Sachverhalt in einem ER-Diagramm ab.

Dazu wird zunächst der Sachverhalt in einem allgemeinen *processmodel* *PM* abgebildet, das alle inhaltlich relevanten Aspekte gemäß des perspektivenorientierten Modellierungskonzept nach [JaBu96] erfasst. In Abbildung 6-11 wird solch ein Prozessmodell als eine Menge von Knoten (= nodes) aufgefasst. Jedem Knoten wird eine bestimmte Elementart zugeordnet, wie zum Beispiel „Prozess“. Darüber hinaus ist jeder Knoten durch Perspektiven charakterisiert, wie zum Beispiel die funktionale oder datenorientierte Perspektive.

Auf Basis dieser Prozessdokumentation kann nun das *variant processmodel* *vPM* erstellt werden. Dies erfolgt durch die Identifizierung und Spezifikation der Prozessvarianten. Es erfolgt die Angabe der *variation points* *vp* und der *variant options* *vo* (Kapitel 6.2.2), wobei die Angabe von *vp* als eine zusätzliche Elementart des Prozessmodells definiert ist, während bei der Angabe von *vo* eine Parametrisierung eines Knoten aus dem Graphen erfolgt. Es muss klar definiert werden, welche *variant option* eines *variation points* zu welcher Variante gehören. Denn einer bestimmten Variante kann nur jeweils eine *variant option* eines *variation point* zugeordnet werden. Eine *variant option* kann jedoch für mehrere Varianten relevant sein, was vor allem bei sehr umfassenden, sehr variantenreichen Prozessmodellen der Fall ist. Meist sind mehrere *variation points* und damit auch *variant options* für eine Variante charakterisierend.

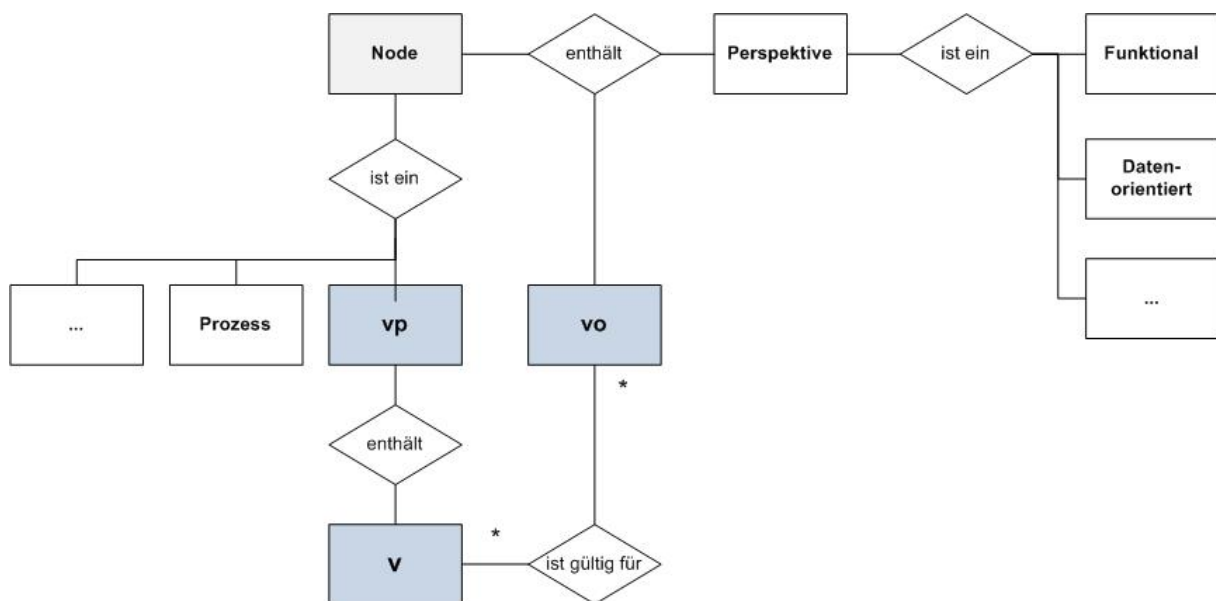


Abbildung 6-11 ER-Diagramm

Aus dem Gesamtprozessmodell gilt es nun die einzelnen Varianten, *process variant* *v*, abzuleiten. Eine konfigurierte Variante enthält schließlich keine *variation points* mehr, sondern besteht nur noch aus nicht-konfigurierbaren Prozesselementen.

6.5.2.2 Prozessverwendung – gestufte Konfiguration

Da ein integriertes Prozessmodell eine Vielzahl von Varianten enthält, die sich nicht immer leicht bzgl. ihrer diversen signifikanten Unterschiede überblicken lassen, soll es möglich sein die gewünschte Variante schrittweise auszuwählen. Letzteres ist auch motiviert durch die Tatsache, dass ein Anwender nach der Prozessmodellierung noch nicht alle wesentlichen Informationen bzgl. des auszuführenden Prozesses zur Hand hat, sondern diese erst nach und nach bekommt.

Daher wird an dieser Stelle die Einführung einer zusätzlichen Phase im Prozesslebenszyklus vorgenommen: die *Prozessvorauswahl*. Diese liegt zeitlich zwischen der *Prozessmodellierung* sowie der sich daran normalerweise direkt anschließenden *Prozessausführung*. Die Prozessvorauswahl ist ein elementarer Bestandteil des hier vorzustellenden *gestuften Konfigurationsprozesses* [CzHE04] [CzHE05]. Die Konfiguration soll zur Prozessvorauswahl wie auch zur Prozessausführung möglich sein.

Der Ansatz der gestuften Konfiguration ist unter anderem auch bei [LaRo08a] und [BeDK07] möglich. Bei [LaRo08a] beziehen sich die Konfigurationsstufen jedoch nur auf den Zeitraum vor der Ausführung und haben primär das Ziel, die syntaktische wie semantische Korrektheit der Modelle nach jedem Konfigurationsschritt sicher zu stellen. Bei dem in dieser Arbeit vorgestellten Konzeptes ist die Konfiguration hingegen auch noch während der Ausführung erlaubt. Zudem sind die Konfigurationsstufen aus dem Aspekt der direkten Anwenderunterstützung bei der Variantenwahl heraus motiviert. Die finale Variante soll inhaltlich für den Anwender adäquat sein. Bei [BKKD01] (oder auch [BeDK07]) sind die einzelnen Konfigurationsstufen bzgl. des Kontextes auf einem höheren Abstraktionsniveau definiert. Zudem ist die Einteilung der Stufen nicht sehr feingranular und fest vorgegeben, so dass die Flexibilität des Ansatzes somit geringer ist. So beinhaltet die erste der beiden Konfigurationsstufen zunächst eine Anpassung an die Domäne bzw. die betreffende Organisation im Allgemeinen, was bei den Modellen im Kontext dieser Arbeit nicht mehr notwendig ist. Danach erst erfolgt in der dritten Stufe die anwendungsfallspezifische Konfiguration, was es im Kontext dieser Arbeit auf mehreren Stufen verteilt erfolgt.

6.5.2.3 Prozessvorauswahl

Im Rahmen der Prozessvorauswahl wird aus dem *variant processmodel vPM*

- eine Teilmenge bestehend aus genau einer einzelnen Variante *process variant v* oder
- eine Teilmenge bestehend aus mehreren Varianten, zusammengefasst in einem partiellen *variant processmodel vPM'*,

ausgewählt. Ersteres ist nur bei ausreichend Informationen über die vorliegenden Kontextbedingungen und die Ausführung möglich. Aufgrund der Komplexität und der Dynamik der heutigen Geschäftswelt ist aber davon auszugehen, dass dies nur sehr selten der

Fall sein wird. Daher kann mit der Wahl einer Teilmenge, die mehrere Varianten umfasst, zunächst eine Vorentscheidung getroffen werden. Sie stellt den Beginn des gestuften Konfigurationsprozesses dar.

Pro variation point können zu diesem Zeitpunkt beliebig viele variant options aus der verfügbaren Menge gewählt werden:

- Es kann nur eine variant option als Teilmenge eines variation points gewählt werden. Dabei sollte daran erinnert werden, dass eine variant option für mehrere verschiedene Varianten relevant sein kann, so dass in diesem Fall noch keine endgültige Entscheidung für eine konkrete Variante gefallen wäre.
- Eine Teilmenge bestehend aus mehr als einer variant options eines variation points kann gewählt werden, da die variant options eines einzelnen variation points sich erst zum Zeitpunkt der Ausführung gegeneinander ausschließen.
- Es werden alle variant options gewählt, wenn der Anwender an dem dazugehörigen variation point zu diesem Zeitpunkt (noch) keine Präferenz bzw. nicht das für die Entscheidung notwendige Wissen hat. In diesem Fall wird keine Teilmenge, sondern die Gesamtmenge an variant options ausgewählt.

Es kann möglich sein, dass gewisse variation points im Rahmen der Prozessvorauswahl entschieden werden müssen, weil der Konfigurationsaufwand während der Ausführung zum Beispiel ansonsten zu groß werden würde. Was die restlichen variation points betrifft, soll der Anwender die Freiheit haben nur die Entscheidungen treffen zu müssen, die er zu diesem Zeitpunkt aufgrund seines Informationsstandes treffen kann bzw. möchte.

Das initiale Gesamtprozessmodell muss nun im Hinblick auf das selektierte Teilmodell konfiguriert und anschließend instanziiert werden. Bei der Konfiguration, auch als Individualisierung bezeichnet, wird folgendermaßen vorgegangen:

- Die nicht mehr relevanten variant options eines gewählten variation points werden gelöscht
- Für den Fall, dass nach diesem Schritt einem variation point nur noch eine variant option zugeordnet werden kann, wird der variation point gelöscht und durch die variant option ersetzt
- Variation point, bei denen keine Wahl getroffen wurde, bleiben unverändert

Die verbleibenden Prozesselemente müssen danach, egal ob eine einzelnen Variante oder eine Teilmenge von Varianten ausgewählt wurde, wieder so miteinander verbunden werden, dass ein konsistentes und valides Prozessmodell entsteht.

Das Ergebnis der Prozessvorauswahl ist entweder eine einzelne *process variant* v oder ein partielles *variant processmodel* vPM' . Letzteres enthält noch eine reduzierte Anzahl von variation points, die im weiteren Verlauf noch aufgelöst werden müssen.

Die Generierung der resultierenden Prozessmodelle stellt keine leichte Aufgabe dar. Selbst bei kommerziellen Lösungen erfolgt sie oftmals noch von Hand [RovA07]. So muss unter anderem überprüft werden, ob die Kontrollflüsse auch die entsprechenden Prozesselemente verbinden; jeder Prozessschritt muss mindestens einen ein- und ausgehenden Kontrollfluss besitzen und der Prozessablauf an sich muss Sinn machen. Zudem darf ein variation point nur definiert sein, wenn diesem mehr als eine variant option zugeordnet sind. Eine *Validierung* des Prozessmodells ist daher essentiell notwendig. Näheres zur Validierung bzw. Überprüfung der Korrektheit von Prozessmodellen ist zum Beispiel [vADG10] zu entnehmen. Hier soll nicht näher darauf eingegangen werden, ohne aber damit die Wichtigkeit dieses Aspektes unterschätzen zu wollen. Eine automatisierte Lösung wäre für die Validierung zu bevorzugen, da diese weniger zeitaufwendig sowie weniger fehleranfällig ist. Die Korrektheit des Transformationsprogrammes zu überprüfen ist dann ausreichend und ist zudem weitaus effizienter zu realisieren [LaVi01] [LiSS94] [ReMA06]. Der Aufwand zur Entwicklung einer automatisierten Lösung sollte jedoch auch nicht unterschätzt werden.

6.5.2.4 Prozessausführung

Nachdem das Prozessmodell instanziiert worden ist, kann mit der Ausführung begonnen werden. Hat sich der Anwender im Rahmen der Prozessvorauswahl für genau eine *variant v* entschieden, wird der Prozess, ohne weitere Aspekte der Konfiguration berücksichtigen zu müssen, ausgeführt. Ist hingegen ein *variant processmodell vPM'* instanziiert worden, müssen weitere Konfigurationsschritte durchgeführt werden. Der Anwender muss sich letztendlich für eine Variante aus dem initialen Prozessmodell entscheiden; da sie sich zum Zeitpunkt der Ausführung gegeneinander ausschließen.

Ein variation point muss spätestens dann aufgelöst werden, wenn dieser bei der Prozessausführung als nächstes auszuführendes Prozesselement auftritt. Je nach bisheriger Konfiguration des Modells kann dann die Entscheidung für eine konkrete Variante fallen oder es müssen noch weitere variation points aufgelöst werden.

Wird sich der Anwender vor der Auflösung aller variation points darüber klar, für welche Variante er sich entscheidet, wird das Prozessmodell zu diesem Zeitpunkt in Bezug auf die gewünschte Variante konfiguriert.

Wie im Rahmen der Prozessvorauswahl muss auch hier nach jedem Konfigurationsschritt, das beim Löschen der nicht relevanten variant options und Auslösen der variation points, eine Validierung des resultierenden Prozessmodells erfolgen, um die Korrektheit des Models zu garantieren.

Im Verlauf der Prozessausführung können theoretisch maximal so viele Konfigurationsstufen realisiert werden, wie es variation points gibt. Doch allein schon aufgrund von wenn-dann-Abhängigkeiten zwischen den variant options werden es meist weniger sein. Die wenn-dann-Abhängigkeiten bewirken, dass bei der Entscheidung eines variation points ein oder mehrere

weitere variation points automatisch mitentschieden werden. Bestimmte Kombinationen von Varianten werden dadurch vermieden, was zum Beispiel auf unternehmenspolitische, organisatorische oder gesetzliche Gründe zurückzuführen ist. Diese Abhängigkeiten müssen in dem Prozessmodell explizit spezifiziert werden.

Es kann sich auch anbieten mehrere variation points bei ihrer Auflösung in zeitlicher Hinsicht zu gruppieren. Auch aus Gründen der Kosteneffizienz sowie des organisatorischen Aspektes sollte die Anzahl auf ein sinnvolles Maß beschränkt werden.

6.5.2.5 Redesign

Es muss davon ausgegangen werden, dass sich die Prozesse zum einen

- *über die Zeit hinweg verändern* (zum Beispiel durch neue/ geänderte Gesetze, durch Erschließung eines neuen Kundenstamms, ...) und zum anderen
- *während der Ausführung Differenzen* zwischen den Vorgaben des Prozessmodells und der tatsächlichen Ausführung auftreten (zum Beispiel durch nicht vorhersehbare Änderungen im Kontext der Prozesse) [vAJa00].

Daher muss es möglich sein, das Prozessmodell zu gegebener Zeit im Rahmen des Starts eines neuen Prozesslebenszyklus zu überarbeiten. Vorher muss stets eingehend geprüft werden, ob auf die während der Ausführung auftretenden Abweichungen mit kurzfristigen Maßnahmen, die nur für den aktuellen Prozesslebenszyklus gültig sind, reagiert werden kann oder ob die Abweichungen langfristigen Charakter haben, die im Zuge des Redesigns in das Modell übernommen werden müssen. Ist Letzteres der Fall, muss sowohl allgemein als auch speziell in Bezug auf Varianten unterschieden werden, ob das Überschreiben der bisherigen Prozessdefinition erfolgen soll, eine Ergänzung des Modells notwendig ist oder Elemente gelöscht werden müssen. In Bezug auf Varianten sind folgenden Design-Maßnahmen möglich:

- Es erfolgt das *Einfügen eines oder mehrerer variation points*. Dazu wird das bereits existierende Prozesselement als variant option gekennzeichnet. Mindestens ein zusätzliches Prozesselement muss außerdem im Zusammenhang mit einem neuen variation point eingefügt werden und als variant option definiert werden.
Zum einen kann Erfahrung das Einfügen von Auswahlmöglichkeiten an einer bestimmten Stelle im Prozess aufzeigen oder aber die Organisation hat sich explizit dazu entschieden, für dieses Merkmal an dieser entsprechenden Stelle Auswahlmöglichkeiten anzubieten, zum Beispiel aufgrund der Überarbeitung des angebotenen Portfolios oder zur Abdeckung unterschiedlicher Kundenwünsche.
- Ein oder mehrere *variant options* werden an einem existierenden variation point *eingefügt*, um das Spektrum an Auswahlmöglichkeiten jeweils zu erweitern.
Die Gründe dafür sind dieselben wie im Zusammenhang mit der Einführung eines neuen variation points (siehe oben), außer dass es hier nicht um die Neuschaffung,

sondern Erweiterung von Auswahlmöglichkeiten geht. Oftmals werden damit Auswahlmöglichkeiten bzgl. unterschiedlichen Qualitätsniveaus definiert, zum Beispiel im Hinblick auf eingesetzte Technologien oder auf Kundenwünsche eingegangen.

- Es kann das *Löschen einer oder mehrerer variant options* erfolgen, wenn diese nicht mehr relevant sind. Dabei müssen mindestens zwei variant options an dem relevanten variation point übrig bleiben.

Dieser Fall kann eintreten, wenn das bis dahin gebotene Ausmaß an Diversifizierung bzgl. der Auswahlmöglichkeiten nicht mehr gefragt ist oder aber durch neue Vorschriften nicht mehr möglich ist. Auch kann es sein, dass die Organisation ihr angebotenes Spektrum aus Kostengründen oder aufgrund von Vorgaben reduziert.

- Ein *variation point* inklusive der variant options bis auf ein verbleibendes Element wird *gelöscht*. Dieses wird als nicht varianten-relevantes Prozesselement in dem Prozessmodell weitergeführt.

Die Gründe für die komplette Reduktion der Auswahlmöglichkeiten sind dieselben wie die für das Löschen einzelner variant options.

- Ein *variation point* wird in einen Entscheidungspunkt *abgeändert*. Diese Änderung beinhaltet auch die Definition der bisherigen Alternativen als variant options. *Umgekehrt* kann ein Entscheidungspunkt mit den zugehörigen Alternativen in einen variation point inklusive variant options umgewandelt werden.

In diesen beiden Fällen hat sich die Bedeutung der Auswahlmöglichkeiten verschoben, so wie es in Kapitel 6.2 erläutert worden ist. Dies kann sich beispielsweise durch geänderte Anforderungen der Anwender oder der Projektbedingungen ergeben.

- Es können ein oder mehrere variant options *inhaltlich* geändert werden.

Da die Anzahl an Auswahlmöglichkeiten gleich bleibt, sind die Gründe dafür in der Anpassung von einzusetzenden Ressourcen (zum Beispiel durch den Einsatz neuer Technologien, geänderte Organisationsform) zu finden.

- Es können ein oder mehrere variant options im Bezug auf die *Zuordnung* zu einer Variante geändert werden.

Diese Änderungen werden aufgrund der Erkenntnis vorgenommen, dass die bisherigen Kombinationen nicht erwünscht sind oder sich aus organisatorischen wie auch ressourcentechnischen Gründen als ungünstig erwiesen haben. Auch können neue Vorschriften, seien es Gesetze oder interne Regelungen, gewisse Kombinationsmöglichkeiten untersagen.

Mit der langfristigen Überarbeitung der Modelle entstehen neue *Versionen* der Prozesse bzw. Prozessmodelle (vgl. Kapitel 6.2.3), unabhängig davon, ob sich die Änderungen an den Modellen auf Varianten oder Alternativen beziehen. Es sollte bedacht werden, dass derartige Änderungen an dem Prozessmodell nicht nach jedem durchlaufenen Prozesslebenszyklus

vorgenommen werden müssen bzw. sollen. Um Aufwand und Kosten zu begrenzen, empfiehlt es sich, dies in regelmäßigen Abständen, zum Beispiel quartalsweise oder halbjährlich, durchzuführen. Eine Ausnahme bilden gravierende Änderungen, wie zum Beispiel aufgrund neuer gesetzlicher Vorschriften, die ab einem bestimmten Zeitpunkt eingehalten werden müssen.

Mit der Möglichkeit des Redesigns schließt sich der Kreis des Prozesslebenszyklus, wie er in Abschnitt 2.3 erläutert wurde, und geht in einen Kreislauf der kontinuierlichen Verbesserung über.

6.5.2.6 Bewertung

Während das Konzept der gestufte Konfiguration an sich und die Idee der variation points bereits in anderen Ansätzen in ähnlicher Form realisiert ist (siehe Kapitel 6.4), ist in Bezug auf diese Arbeit hervorzuheben, dass der Anwender den Zeitpunkt der Auflösung prinzipiell frei wählen kann und auch noch während der Ausführung die Möglichkeit zur Konfiguration hat. Dies ist dann von Vorteil, wenn der Anwender erst zu diesem Zeitpunkt die für die Entscheidung notwendigen Informationen vorliegen hat. Falsche Entscheidungen bzgl. einer Variante, die aufgrund unzureichender Informationen vor der Ausführung getroffen werden, können damit vermieden werden. Es kann zudem ein kompletter Wechsel zwischen Varianten zur Ausführungszeit vermieden werden, wie er beim Provop-Ansatz vorgesehen ist, da dies einen großen organisatorischen wie auch implementierungstechnischen Änderungsaufwand mit sich bringt. Im Gegenzug können Anwender variantenbezogene Entscheidungen unabhängig von deren Position im Prozessablauf gleich zu Beginn des Konfigurationsprozesses treffen, so dass essentielle Eigenschaften der gewünschten Variante auf jeden Fall Bestandteil der finalen Variante sind. Damit bietet ihm dieser Ansatz ein Höchstmaß an Flexibilität.

Mit dem gestuften Konfigurationsprozess, der sich über mehrere Phasen des Prozesslebenszyklus erstrecken kann, ist zwar das Management der Varianten erleichtert. Das Problem der strukturierten Modellierung in einem und demselben Modell stellt sich aber nachwievor. Diesbezüglich wurde als Kernpunkt des vorliegenden Ansatzes das Konzept einer Datenbasis entwickelt. Darauf wird nun in den folgenden Abschnitten eingegangen.

6.5.3 Anforderungen an die konfigurierbare Datenbasis

Als Basis für den Variantenmanagementansatz ist eine Datenbasis notwendig, in der die Varianten abgespeichert werden können. Auf diese wird dann im Rahmen der Prozesskonfiguration zugegriffen (siehe Abbildung 6-10). Folgende Anforderungen wurden daher für die Datenbasis ermittelt:

- Die Datenbasis soll nicht nur alle Varianten, die im Rahmen der Konfiguration ausgewählt werden können, enthalten, sondern auch alle anderen Prozesse des

jeweiligen Anwendungsfalls in einem Modell integrieren (*single model approach*) [HaBR10] [ReMT09] [RovA07]. Dieser Ansatz kann somit mit dem Konzept eines Referenzmodells, das als Repositorium für empfohlene Vorgehensweisen innerhalb einer bestimmten Domäne zu sehen ist [FeLo03] [Schü98] [vBro03], verglichen werden. Die Verwendung eines solchen Repositoriums beschleunigt den Konfigurationsprozess, weil auf existierende Elemente zurückgegriffen werden kann [GAVL07] [HaBR10] [ReMT09] [RovA07].

- Entscheidend ist die Funktionalität zur *Dekomposition* der (Prozess-) Modellelemente. Dies ermöglicht erst die Darstellung mehrerer Varianten einer Domäne in einem integrierten Modell [MiFr89] [SaWe97].
- Die Datenbasis muss zum einen ermöglichen verschiedenste Komponenten und deren Beziehungen, die diese untereinander haben, zu erfassen. Darauf aufbauend müssen die gewünschten Einzelobjekte, hier die einzelnen Prozessvarianten abgeleitet werden können, was im Sinne einer Spezialisierung zu sehen ist [Bode06] [MiFr89] [SaWe97].
- Der modellbasierte Ansatz wird außerdem durch den Aspekt der *Wiederverwendung* motiviert; jede konfigurierte Variante wird dann als eine individuelle Kombination (= Komposition) der zuvor in der Datenbasis definierten Elemente definiert [BDDK04] [SaWe97] [TiSo97].
- Jede einzelne Variante innerhalb des integrierten Modells muss wieder aus dem integrierten Modell rekonstruierbar sein.
- Das zu konfigurierende Objekt, hier die Prozesse bzw. das dazugehörige Prozessmodell, muss die *Struktur* der Datenbasis beschreiben können. Eine solche (Daten-) Struktur ist essentiell für das Management sowie die Konfiguration solcher Objekte [RMTF99] [RovA07] [SaWe97] [SmDu01].
- Die *verschiedenen Aspekte* der Prozesse (zum Beispiel der funktionale, operationale oder organisatorische Aspekt) müssen in der Datenbasis abgebildet werden können. Nur wenn alle relevanten Aspekte berücksichtigt werden, ist eine umfassende Beschreibung der zu konfigurierenden Prozesse möglich [CuKO92] [JaBu96].
- Es muss ein geeigneter Variabilitätsmechanismus vorhanden sein, um aus einem integrierten Prozessmodell individuelle Varianten ableiten zu können. Diese Anforderung ist in Anlehnung an die erfolgreichen Variantenmanagementkonzepte im Bereich der Software Produktlinien (siehe dazu zum Beispiel [CINo02], [SvBo99] oder [PBvL05]) zu sehen.
- Entsprechend der Forderung nach einem Variabilitätsmechanismus soll das integrierte Prozessmodell *konfigurierbar* sein [SvBo99]. Dies beinhaltet, dass das Modell eine entsprechende Funktionalität zur Anpassung an bestimmte Kontextbedingungen besitzt, so dass die erforderlichen und/oder gewünschten Prozesselemente ausgewählt werden können.

- Um zu erkennen, an welchen Stellen im Prozessmodell bzw. in Bezug auf welchen Aspekt die Unterschiede zwischen den einzelnen Varianten vorhanden sind, soll das Konzept der *variation points* verwendet werden.
- Mit der Definition der *variation points* geht einher, dass klar sein muss, was die Unterschiede ausmacht, d.h. welche Wahlmöglichkeiten (*variant options*) im Hinblick auf die verschiedenen Varianten vorhanden sind.
- Das in der Datenbasis enthaltene Prozessmodell muss in einer *verständlichen, lesbaren sowie verwendbaren* Art und Weise aufgebaut und gespeichert sein.
- Eine möglichst *kompakte und übersichtliche Darstellung* der Varianten sollte realisiert werden.
- Es ist nicht ausreichend die konfigurierbaren Prozesselemente bzw. *variant options* an sich darzustellen. Vielmehr müssen auch *Abhängigkeiten bzw. Einschränkungen* zwischen ihnen abgebildet werden können, um den Variantenraum somit gezielt begrenzen zu können [SaWe97]. Aus rechtlichen, organisatorischen aber auch aus technischen Gründen können gewisse Kombinationen an Prozesselementen explizit vorgegeben oder aber verboten sein. Dies betrifft meist Teilbereiche der Variante, selten eine Variante im Ganzen. Neben der bereits geforderten Kompaktheit ist damit auch die Abbildung der realen Prozesswelt möglich. Es sollte angemerkt werden, dass es gilt, solche Abhängigkeiten bzw. Einschränkungen auch in Bezug auf normale Entscheider und Konnektoren zu berücksichtigen.
- Die Funktionalität zur Abbildung der Variabilität und Realisierung eines Konfigurationsmechanismus sollte *losgelöst von einer bestimmten Prozessmodellierungssprache* erfolgen, um ein möglichst breites Einsatzgebiet für den gesamten in dieser Arbeit konzipierten Variantenmanagement-Ansatz zu erschließen.

Wie diese Anforderungen nun umgesetzt werden, wird im folgenden Abschnitt erläutert.

6.5.4 Das Konzept der konfigurierbaren Datenbasis

Die Summe der in Kapitel 6.5.3 ermittelten Anforderungen lässt sich nur schwer in einem einzigen Modell umsetzen, da zum einen eine benutzerfreundliche zugleich aber auch mächtige und aussagekräftige Darstellung des integrierten, konfigurierbaren Prozessmodells angestrebt wird.

Daher erfolgt die Anwendung eines Konzeptes auf die *Datenbasis*, wie sie in Abbildung 6-12 zu sehen ist und im Folgenden präsentiert wird¹⁰. Entscheidend ist, dass ein und derselbe Inhalt eines Modells auf verschiedenen Art und Weisen dargestellt werden kann (siehe dazu auch [Götz10] und [Volz11]). Die hier vorzustellenden Datenbasis sieht zum einen die

¹⁰ Die Lesbarkeit der beiden Modellinhalte in Abbildung 6-12 ist nicht entscheiden, sondern wie sie konzeptuell miteinander zusammenhängen.

Darstellung des Modellinhaltes in einem benutzerfreundlichen *Anwender-orientierten Prozessmodell* vor, zum anderen gilt es die Darstellung für ein prozessorientiertes, jedoch abstraktes bzw. *generisches Datenmodell* zu definieren, das die Variabilität präzise spezifiziert. Das damit verfolgte Hauptziel ist, die Komplexität, die mit der Modellierung sowie Konfiguration von Varianten in einem integrierten Prozessmodell unweigerlich verbunden ist, in Teilen vor dem Anwender zu verbergen.

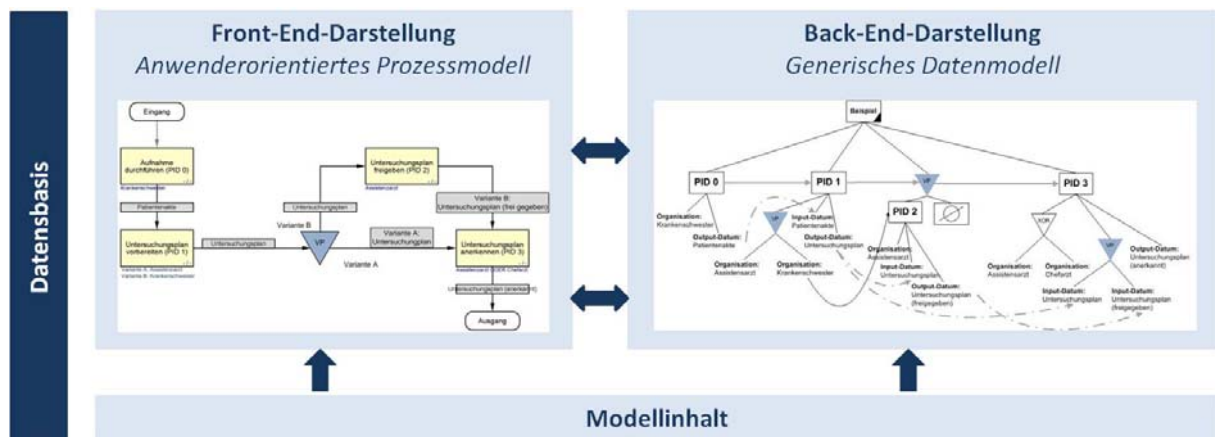


Abbildung 6-12 Konzeptueller Aufbau der Datenbasis

Das anwenderorientierte Prozessmodell

Das anwenderorientierte Prozessmodell stellt die Prozesse und seine Varianten in einer beliebigen Prozessmodellierungssprache, wie zum Beispiel AOPM oder BPMN, dar. Es dient zur Präsentation für die Prozessausführenden bzw. -Verantwortlichen und wird somit im sog. *Front-End*, der Schnittstelle zum Benutzer des Prozesskonfigurators, verwendet bzw. angezeigt. Der Schwerpunkt liegt auf einer benutzerfreundlichen Darstellung der Prozessmodelle und deren Inhalte.

Das generische Datenmodell

Im Hinblick auf das generische Datenmodell wurde ebenfalls eine bildliche Darstellung gewählt. Der Schwerpunkt liegt auf der Darstellung der Variabilität der Prozesse, so dass die entsprechende Funktionalität in der Modellierungsnotation integriert sein muss. Das generische Datenmodell soll nicht dem Anwender des Prozesskonfigurators präsentiert werden, sondern ist Bestandteil des sog. *Back-Ends*, der Basis für die informationstechnische Umsetzung.

Es bietet sich die graphische Notation in Form einer *Teile-Ganzes-Beziehung* an bzw. die eines Graphen bzw. Baumes, der eine hierarchische Darstellung eines Sachverhaltes ermöglicht [WeMü81]. Bei Bäumen handelt es sich im Allgemeinen um eine sehr ausdrucksstarke Darstellungsform für konfigurierbare Objekte (siehe auch [Ehrl07] oder [Hein99]). Sie

werden sehr oft zur logischen Darstellung hierarchischer Strukturen und den dazugehörigen Varianten verwendet. Grund ist der hohe Grad an Abstraktion, Kompaktheit sowie Übersichtlichkeit. Die im Rahmen dieser Arbeit vorgenommene Definition des generischen Datenmodells wurde motiviert durch die Arbeiten [Gair81], [RMTF99] sowie vor allem [Hümm04], in denen das Konzept der Bäume zur Beschreibung konfigurierbarer Produkte bzw. Serviceleistungen verwendet wird. Die Idee ist, diese Konzepte bzw. diese Ansätze auf den Bereich der Prozessmodellierung und die Konfiguration von Prozessvarianten zu übertragen. Das konzipierte generische Datenmodell basiert letztendlich primär auf dem sog. *mereologischen Graphen* nach [Hümm04], der eine besondere Form einer Variantenstückliste darstellt. Dieser Graph hat als wesentlichen Bestandteil das Element der Alternative, das eine effiziente und übersichtliche Formulierung einer Menge von Varianten in einem einzigen Graphen erlaubt.

6.5.5 Das Generische Datenmodell

Das generische Datenmodell basiert primär auf dem von Hümmel [Hümm04] (siehe auch in Kurzform Englisch in [HMMD04]) für die Konfiguration von Produkten erweiterten mereologischen Graphen. Diese besondere Form der Variantenstückliste wurde zunächst von [WeMü81] vorgestellt. In das nun vorzustellende Konzept des Datenmodells sind bereits Ergebnisse einer Evaluation eingeflossen. Über diese wird in Kapitel 6.5.7 berichtet.

6.5.5.1 Allgemeine Definition

Die wesentlichen Bestandteile des generischen Datenmodells sind: *Knoten*, *Kanten* und *Verbindungen*. Die Kanten verbinden Knoten als Ober- und Unterteile miteinander, Verbindungen definieren gerichtete Beziehungen zwischen den Knoten. Mit diesen Elementen gilt es nun die diversen Aspekte einer Prozessmodellierungssprache zu erfassen und zugleich eine Konfigurierbarkeit des Modells zu ermöglichen. Für die Verwendung des Ansatzes im Kontext des Variantenmanagements von Prozessen wurde der mereologische Graph in dieser Arbeit entsprechend angepasst. Folgendes wurde definiert (ein dazugehöriges Beispiel ist in Abbildung 6-13 zu sehen, als Basis wurde das Beispiel aus Abbildung 6-18 verwendet):

- *Prozesse* werden als Knoten dargestellt. Komposite Prozesse, in Abbildung 6-13 Rechtecke mit einer schwarzer Ecke, fassen mehrere Einzelprozesse auf einem höheren Abstraktionslevel zusammen. Solch eine Komposition kann wiederholt durchgeführt werden, was zur Bildung von Hierarchieebenen bzw. zu einer Baumstruktur führt. Die Wurzel des Graphen muss immer ein kompositers Prozess sein. Elementare Prozesse werden in Form von einfachen Rechtecken ohne ausgefüllte Ecke dargestellt und sind nicht weiter zerlegbar.

- Zwischen den einzelnen Prozessschritten kann optional explizit ein *Kontrollfluss* definiert werden. Er leitet sich ab aus der durch den Datenfluss vorgegebenen Reihenfolge der einzelnen Prozessschritte und dient der Verbesserung von Lesbarkeit und Verständnis des Modells. Der Kontrollfluss wird durch einen grau gestrichelten Pfeil dargestellt, wie er in Abbildung 6-13 exemplarisch zwischen Prozess 0 und 1 angezeigt ist.
- *Alternativen (OR)* und *exklusive Alternativen (XOR)* stellen Knoten dar und werden graphisch als ein auf der Spitze stehendes Dreieck mit weißer Füllung und schwarzem Rand modelliert. Von den zugeordneten Prozessen werden bei der späteren Ausführung entweder eine Teilmenge an Prozessen oder aber genau ein Prozess gewählt. Eine *Konjunktion (AND)* steht für die unabhängige Ausführung der daran angehängten Prozesse.

Die bisherige Definition führt zu einer grundlegend *funktionalen Strukturierung* des zugrunde gelegten Datenmodells. Dies stellt eine gängige Methode zur Komplexitätsbeherrschung konfigurierbarer Objekte dar [MiFr89] [Riit01] [RovA07]. Eine Funktion ist eine wesentliche bzw. essentielle Eigenschaft eines jeden Objektes; über die Funktionen des Objektes können die jeweils nötigen Aspekte/ Komponenten für das finale Objekt gezielt abgeleitet werden [MiFr89].

Darüber hinaus ist Folgendes festgelegt:

- Das Konzept des generischen Datenmodells wird nun durch weitere *aspektorientierte Knoten* bzw. genauer gesagt Blätter ergänzt (der funktions- bzw. verhaltensorientierte Aspekt sind davon ausgeschlossen). Sie werden mit einem Prozesselement, sei es elementar oder komposit, über eine Kante verbunden, d.h. diesem untergeordnet. Die am häufigsten definierten Aspekte sind der organisatorische, operationale sowie datenorientierte Aspekt [BAKF04] [CuKO92] [JaBu96]. Weitere Aspekte sind denkbar und können jederzeit hinzugenommen werden. Prinzipiell kann jegliche Art von Information, die im Hinblick auf einen Prozess als essentiell erachtet wird, auf dieser Art und Weise spezifiziert werden. Wie in Abbildung 6-13 zu sehen ist, werden solche nicht-funktionalen Aspekte graphisch als Textfelder ohne Umrandung dargestellt; in den Textfeldern ist der jeweilige Aspekt fest definiert und die konkrete Ausprägung wird individuell angegeben. Jedem aspektorientierten Knoten können wiederum aspektorientierte Knoten untergeordnet werden, wodurch sich eine weitere Hierarchisierung des Modells ergibt.
- Bei dem *datenorientierten Aspekt* wird zwischen Input- und Output-Daten der Prozessschritte unterschieden.

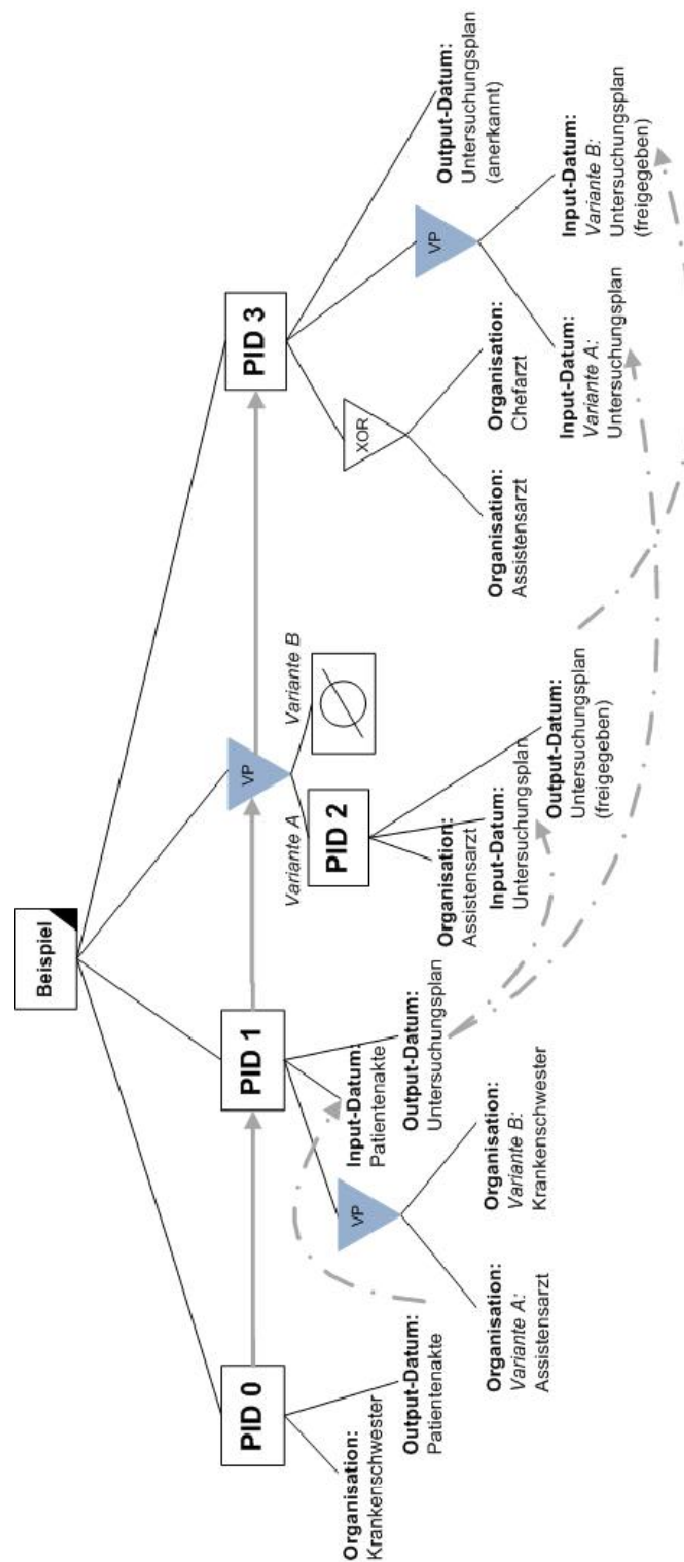


Abbildung 6-13 Generisches Datenmodell [eigene Quelle]

- Um genau abbilden zu können, welche Daten von einem Prozessschritt produziert und vom welchem nachfolgenden Prozessschritt diese konsumiert werden, wird ein expliziter *Datenfluss* als Verbindungselement zwischen Output- und Input-Datum definiert. Die graphische Darstellung erfolgt mittels eines grau gestrichelten Pfeils. In Abbildung 6-13 wird zum Beispiel vom Output-Datum „Patientenakte“ des Prozesses PID 0 zum Input-Datum „Patientenakte“ des Prozesses PID 1 ein Datenfluss angezeigt.
- Im Bezug auf die verschiedenen aspektorientierten Knoten werden die bisher nur in Bezug auf den verhaltensorientierten Aspekt betrachteten Konjunktionen und Alternativen zu *aspektorientierten, logischen Konjunktionen* sowie *Alternativen* erweitert. Diese können sich damit auch auf dem datenorientierten, organisatorischen Aspekt usw. beziehen. Beispielhaft zu sehen ist dies in Abbildung 6-13 bei PID 3, wo dies am Beispiel des organisatorischen Aspektes veranschaulicht wird. Die graphische Darstellung erfolgt, wie auch bei den verhaltensorientierten Konnektoren, als ein auf der Spitze stehendes Dreieck mit weißer Füllung und schwarzem Rand.
- Um zwischen optionalen und obligatorischen Alternativen bei den verschiedenen Aspekten unterscheiden zu können, wird ein weiterer Knoten als sog. *leeres Element* verwendet [Hümm04]. Für dieses Element ist kein Inhalt vorgesehen. Dargestellt wird es als Rechteck mit durchgestrichenem Kreis.
- Da die Entscheidungen von verschiedenen Alternativen (siehe Kapitel 6.2.2) nicht immer unabhängig voneinander sind, ist die Einführung von sog. *Implikationen* sinnvoll. [Hümm04] verwendet die Implikationen in Bezug auf Varianten, was hier nicht erfolgen soll; für diese wird separat ein entsprechendes Konzept definiert. Implikationen stellen eine Beziehung zwischen ein oder mehreren *Vorbedingungen* (= *Antezedens*) und einer *Folgerung* (= *Konsequenz*) her. In der natürlichen Sprache wird eine solche Beziehung als „wenn...dann“ formuliert. Unabhängige Entscheidungsspielräume, hier zwei voneinander unabhängige Alternativen, können damit organisiert werden. Der Entscheidungsraum kann damit gezielt eingeschränkt und die Anzahl der möglichen Pfade durch das Prozessmodell effizient reduziert werden [SaWe97].

Die Darstellung erfolgt durch einen von dem Antezedens zur Konsequenz zeigenden schwarzen Pfeil, wie es in Abbildung 6-14 zu sehen ist. Die per se richtungsabhängig definierten Implikationen werden aus implementierungstechnischen Gründen bzw. zur Vereinfachung der Implementierung in Richtung des Daten- bzw. Kontrollflusses eingetragen. Dabei ist es erlaubt, *Abhängigkeiten zwischen verschiedenen Aspekten* zu definieren, zum Beispiel zwischen dem organisatorischen und funktionalen Aspekt wie es in Abbildung 6-14 zu sehen ist. Dies bedeutet, dass Vorbedingung und Folgerung nicht zu demselben Aspekt gehören müssen. Es spielt auch keine Rolle, ob es sich dabei um OR- oder XOR-Konnektoren handelt. Die Definition einer

Implikation erfolgt bei Bedarf, sie muss nicht für jeden Entscheidungspunkt angegeben werden.

Es können verschiedene Arten von Implikationen unterschieden werden: Als einfachster Fall können 1:1-Abhängigkeiten definiert werden (elementare Implikation), bei denen genau zwei Elemente eines Prozesses, d.h. Knoten, verknüpft werden (siehe Abbildung 6-14). Darüber hinaus besteht die Möglichkeit 1:n-Abhängigkeiten zu modellieren (erweiterte elementare Implikation). Dabei zieht die Auswahl eines Prozesselements an einem Konnektor eine Entscheidung an mehreren weiteren Konnektoren nach sich. Die Analyse der Beispielgraphen aus den Lehrstuhlprojekten hat ergeben, dass eine n:1-Abhängigkeit sehr selten auftritt. Die Definition einer n:m-Abhängigkeit ist ebenfalls denkbar (komplexe Implikation), ist in den Lehrstuhlprojekten jedoch nicht aufgetreten. Eine graphische Darstellung ist mangels Eindeutigkeit zudem nicht mehr möglich (siehe [Hümm04]), daher wird sie für den Kontext der Prozesse ausgeschlossen.

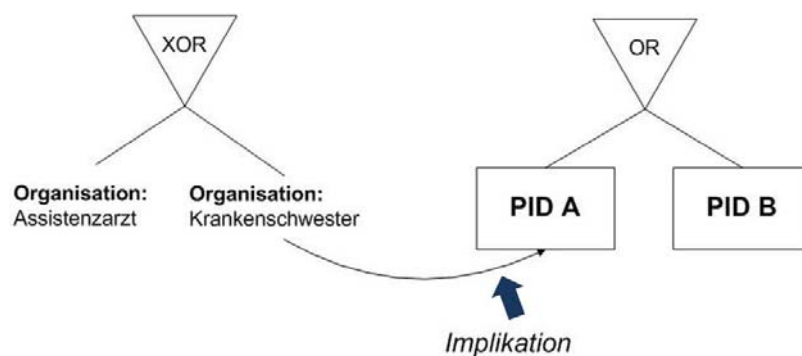


Abbildung 6-14 Implikation zwischen verschiedenen Aspekten

Nach dieser allgemeinen Beschreibung des prozessorientierten Datenmodells wird nun speziell auf die Elemente eingegangen, die die Konfigurabilität ausmachen.

6.5.5.2 Variation points

Mittels eines Knotens als ein blau ausgefülltes, auf der Spitze stehenden Dreieckes mit dem Label *VP* (siehe Abbildung 6-15 bzw. Abbildung 6-13 im Gesamtkontext eines Prozesses) erfolgt explizit die graphische wie auch logische Darstellung eines *variation points*.

Solch ein *variation point* kann als Unterteil eines kompositen oder elementaren Prozesses sowie eines verhaltens- sowie nicht verhaltensorientierten Konnektors definiert werden. Er muss selber stets mindestens zwei untergeordnete Knoten besitzen, wodurch ein *variation point* zugleich die Funktion eines Oberteils einnimmt.

Die solch einem variation point untergeordnete Knoten stellen die *variant options* dar. Sie können für jeden beliebigen Aspekt definiert werden. Sie müssen aber auf einen variation point bezogen sein und demselben Aspekt zugeordnet sein, so wie es in Abbildung 6-15 zum Beispiel der organisatorische Aspekt ist.

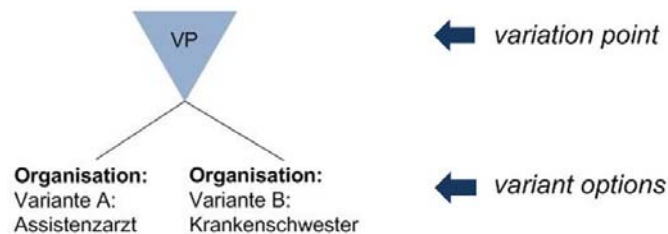


Abbildung 6-15 variation point und variant options

Handelt sich bei den variant options um Prozesse, so können der dazugehörige variation point oder, wenn dies die Eindeutigkeit erfordert, die variant options selber mit ein- und ausgehenden *Kontrollflüssen* modelliert werden. Sind Datenelemente einem variation point zugeordnet, werden die *Datenflüsse* immer direkt an den variant options modelliert. Jede variant option muss dann mindestens einen ein- oder einen ausgehenden Datenfluss besitzen.

Bei jeder variant option wird angegeben, welcher Variante diese zuzuordnen ist. Dies erfolgt mittels eines *textuellen Zusatzes* an den einzelnen variant options (siehe Abbildung 6-13), die die *Bezeichnung* der jeweiligen Variante anzeigen. Während hier eine alphabetische Bezeichnung vorgenommen wurde, kann ebenso eine Nummerierung verwendet werden; auch eine namentliche Bezeichnung ist denkbar. Sinn und Zweck der Bezeichnung ist vergleichbar mit denen der Implikationen bei Entscheidungspunkten bzw. Alternativen (siehe Abbildung 6-14). Die Entscheidungsspielräume, hier als Variantenraum bezeichnet, und die Anzahl der möglichen Varianten sollen effizient beschränkt werden; Korrektheit und Relevanz der Varianten können garantiert werden.

Um die Unterscheidung zwischen variation points und Entscheidungspunkt (siehe Kapitel 6.2.2) konsequent beizubehalten, wurden hier bewusst nicht die von [Hümm04] verwendeten Implikationen wiederverwendet. Sie würden zum einen zur Darstellung der Kombinationsmöglichkeiten der variant options den Graphen sehr unübersichtlich machen, da sie an jeder variant option definiert werden müssten. Die Implikationen sind im Kontext dieser Arbeit außerdem aus implementierungstechnischen Gründen in Richtung des Daten- bzw. Kontrollflusses definiert. Aufgrund des gestuften Konfigurationsprozesses (siehe Kapitel 6.5.2) werden die variation points jedoch nicht notwendigerweise entlang des Kontrollflusses aufgelöst. Sind zwei variation points mit einer Implikation verbunden, bedeutet die Wahl des im Prozessablauf später definierten variation points die Wahl einer Konsequenz. Es müsste

dann sichergestellt werden, dass das dazugehörige Antezedens implizit mit gewählt wird, was einen zusätzlichen Validierungsschritt erfordert.

Es ist möglich bzw. erlaubt, die variation points *hierarchisch anzuordnen*. Damit werden „Varianten in einer Variante“ definiert. Ein variation point wird dabei selber einem variation point untergeordnet. Dies kann auf direkt darunter liegender Ebene erfolgen; der untergeordnete variation point wird somit zugleich eine variant option. Vor allem bei sehr umfassenden Prozessen erfolgt hingegen die Definition eines solchen hierarchischen variation points oftmals nicht auf der direkt darunter liegenden Ebene, sondern weitaus tiefer, wie es in Abbildung 6-16 an einem Beispiel aus dem medizinischen Lehrstuhlprojekt zu sehen ist (auf inhaltliche Aspekte soll es hier nicht ankommen, daher ist die Darstellung sehr klein). Jeder variation point innerhalb dieser Hierarchie kann dabei in Bezug auf einen anderen Aspekt definiert sein. Somit kann zum Beispiel eine Variante in Bezug auf den funktionalen Aspekt durch eine Variante in Bezug auf den organisatorischen Aspekt verfeinert werden. Ein Beispiel dafür wären die Prozesse der elektiven Aufnahme und der Notfallaufnahme in einem Krankenhaus, wobei bei Ersterem noch unterschieden werden kann, ob ein Arzt oder eine Krankenschwester/ein Pfleger gewisse Schritte durchführen. Mit der Auflösung der variation points kann auf jeder Ebene begonnen werden. Wird ein variation point auf einer oberen Hierarchieebene aufgelöst, kann dies gegebenenfalls die Löschung eines noch unaufgelösten variation points innerhalb der nicht-relevanten variant option zur Folge haben. So verhält es sich zum Beispiel bei der Wahl des leeren Elementes beim rechten markierten Beispiel in Abbildung 6-16. Mittels der hierarchischen Modellierung der variation points kann die Menge der variant options bzw. die der Varianten noch einmal zusätzlich strukturiert werden.

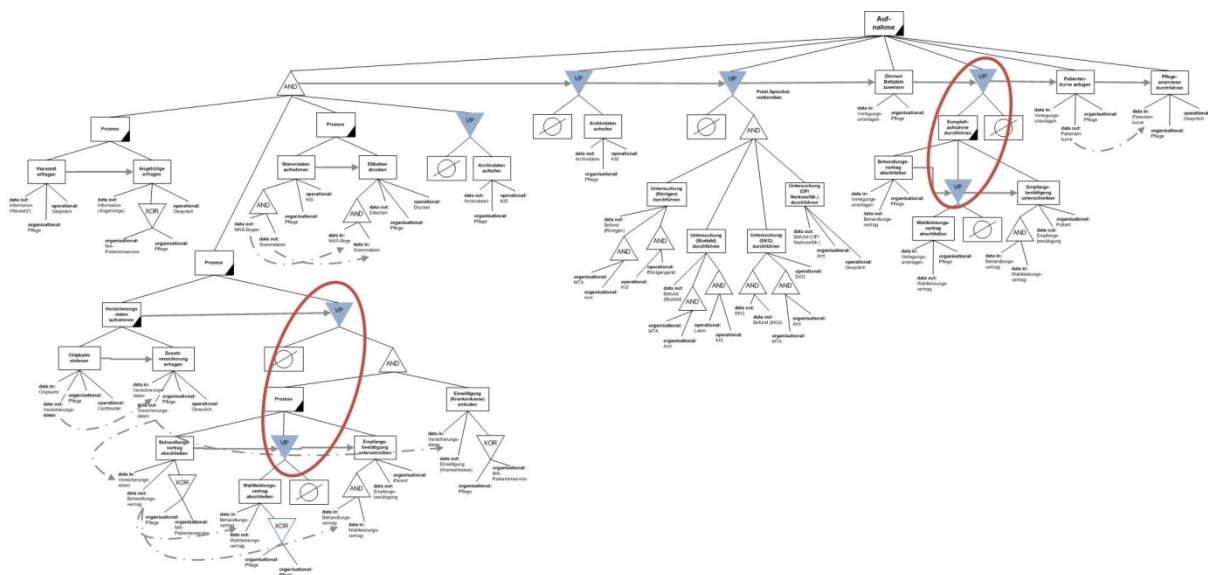


Abbildung 6-16 Hierarchisch angeordnete variation points [eigene Quellen]

6.5.5.3 Anwendungsbeispiel

Das folgende Beispiel zeigt, wie bei der Auflösung der variation points die Abhängigkeiten der variation options den Konfigurationsprozess im Hinblick auf die Anzahl der Konfigurationsstufen beeinflussen können.

Tabelle 6-1 zeigt in der linken Spalte schematisch die variation points und variant options eines fiktiven Prozessmodells. Insgesamt sind vier Varianten A, B, C und D in dem Modell integriert. Die Nummerierung der variation points entspricht in diesem Beispiel der Reihenfolge ihres Auftretens entlang des Prozessverlaufs, wobei es sich damit um ein sehr einfaches Beispiel handelt. Die Reihenfolge kann bei anderen Modellen, bei denen aufgrund von Konjunktionen und Alternativen nicht immer derart eindeutig bestimmt werden.

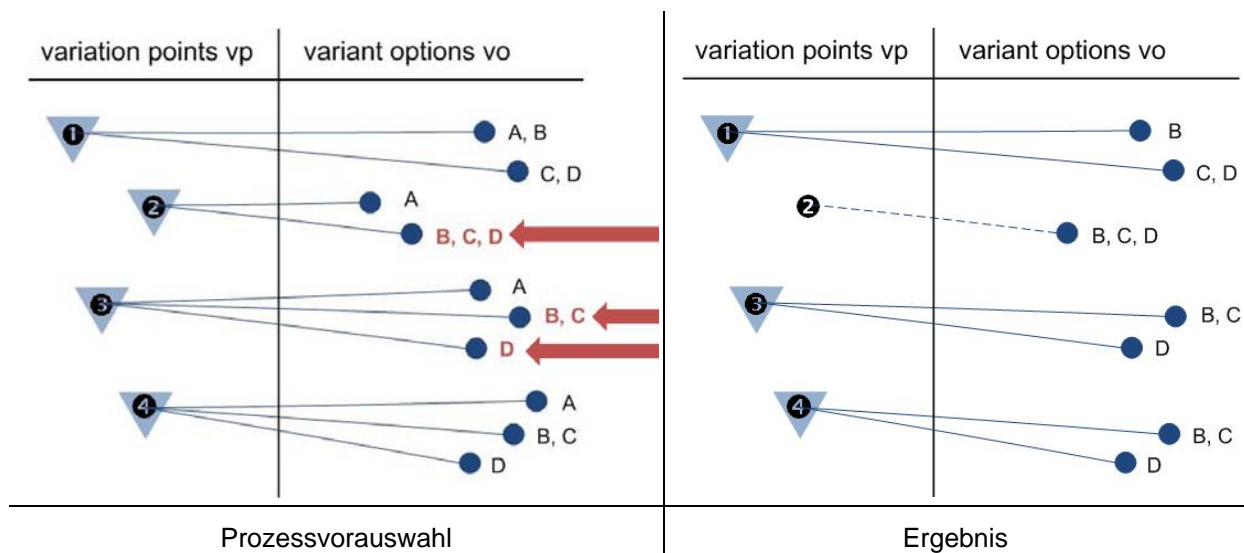


Tabelle 6-1 Prozessvorauswahl – Auswahl der variant options

Die durch die dicken Pfeile gekennzeichneten variant options werden vom Anwender in der ersten Konfigurationsstufe ausgewählt. Das Ergebnismodell ist in der rechten Spalte zu sehen. Variante A ist nach diesem Konfigurationsschritt nicht mehr wählbar; in Bezug auf die restlichen Varianten ist noch keine Entscheidung getroffen. Nun ist für die noch folgende Anzahl an Konfigurationsstufen entscheidend, welche variant options der Anwender wählt.

Wählt der Anwender die variant option D an variation point 3 ist die Konfiguration beendet, da eine eindeutige Entscheidung bzgl. Variante D gefallen ist. Tabelle 6-2 zeigt die Auswahl sowie das Ergebnis. Selbiges gilt ebenso für variation point 4, wenn dort variant option D gewählt wird.

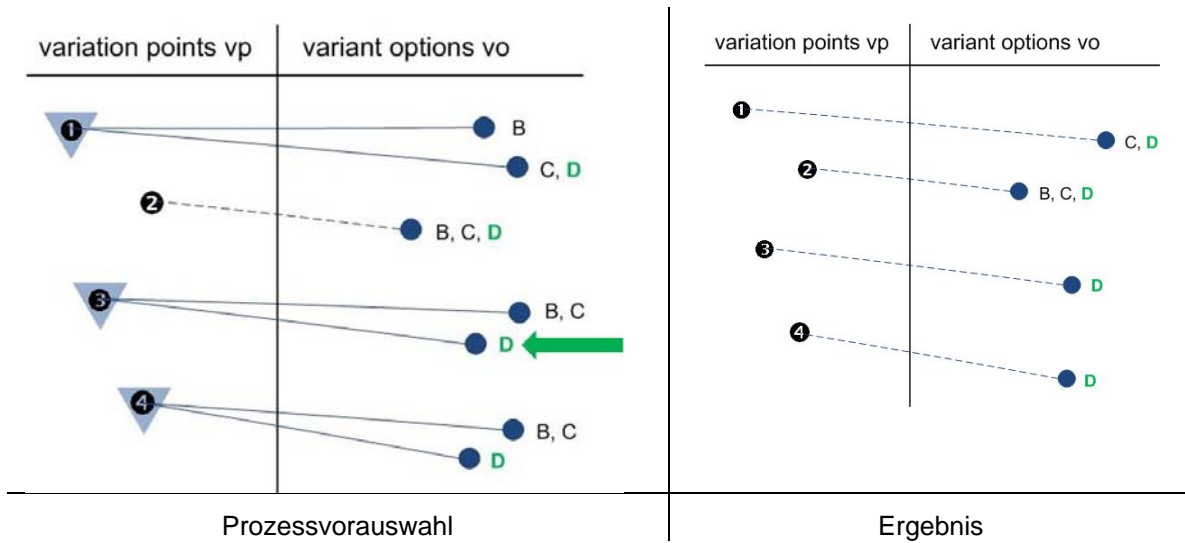


Tabelle 6-2 Prozessausführung - Entscheidungsmöglichkeit I

Wählt der Anwender hingegen bei variation point 3 die mit B und C gekennzeichnete variant option (Tabelle 6-3), ist jeweils noch ein weiterer Auswahlschritt notwendig, um sich letztendlich zwischen Variante B, C und D entscheiden zu können.

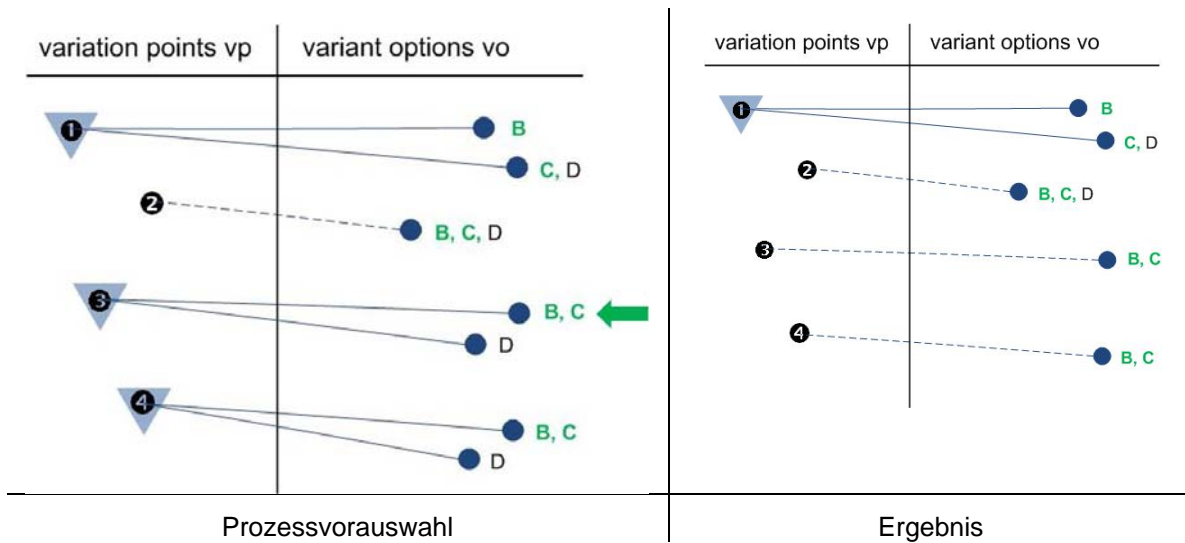


Tabelle 6-3 Prozessausführung - Entscheidungsmöglichkeit II

6.5.5.4 Bewertung

Durch die bildliche Darstellung können die komplexen Zusammenhänge in dem generischen Datenmodell explizit und anschaulich abgebildet werden. Die Darstellung ist bzgl. des Inhalts sehr flexibel, da sie jederzeit ohne großen Aufwand erweiterbar ist; an den einzelnen Elementen können stets beliebig weitere Arten von Unterteilen bzw. Blättern angehängt werden. Bei einem Dokument (datenorientierter Aspekt) wäre zum Beispiel die zusätzliche Angabe des Erstellungsdatums denkbar.

Das Konzept bzw. Modellierungselement des variation points ermöglicht das gezielte Ableiten von einzelnen Varianten aus dem integrierten Modell. Durch die explizite Modellierung der variation points, die nicht auf einen bestimmten Aspekt eines Prozesses fokussiert sind, kann jegliche Art von Prozessvariabilität erfasst werden, d.h. jedes Prozesselement, wie zum Beispiel Funktionen, Daten oder Organisationen, kann als konfigurierbar erfasst werden. Dies erfolgt durchgängig in einer einheitlichen Darstellungsweise, was die Verständlichkeit und Lesbarkeit des Datenmodells positiv beeinflusst. Die variation points lassen sich zudem wie jedes andere Modellierungselement als Ober- bzw. Unterteil in das Modell einbinden, ohne dass besondere Schritte beachtet werden müssen oder an anderer Stelle noch Zusatzinformationen abgelegt werden müssen, wie es zum Beispiel bei [HaBR10] mit der separaten Angabe der Optionen zu den Aufsetzpunkten der Fall ist (siehe auch Kapitel 6.4.3). Für die dazugehörigen variant options müssen keine zusätzlichen Modellierungselemente definiert werden, da bei Verwendung der vorhandenen Prozesselemente ihre Bedeutung durch die Zuordnung zu den variation points eindeutig spezifiziert ist. Es wird lediglich über ein optionales Attribut die Zugehörigkeit zu einer bestimmten Variante angegeben, wie es in Abbildung 6-17 an einem Modellausschnitt zu sehen ist. Dieses Vorgehen reduziert die Anzahl der Modellierungselemente und wirkt sich positiv auf den sich anschließenden Implementierungsaufwand aus.



Abbildung 6-17 Darstellung der konfigurierbaren Stellen

Als Hauptvorteil ist die *Sprachunabhängigkeit* des generischen Datenmodells zu sehen. Über die nur als beispielhaft anzusehende Anwendung von AOPM hinaus können die

Grundelemente des generischen Datenmodells (Knoten, Kanten, Verbindungen inklusive deren verschiedener Ausprägungen) auch auf andere Prozessmodellierungssprachen abgebildet werden. In Tabelle 6-4 ist dies neben POPM beispielhaft an Hand von UML (Aktivitätsdiagramm), EPKs und BPMN demonstriert.

Die variation points und variant options tauchen hier bei den Knoten nicht in einer eigenen Zeile auf. Die variation points gilt es explizit nur in dem generischen Datenmodell zu definieren; ein Knoten (bzw. Prozesselement) hat die Eigenschaft einer variant option inne, sobald er einem variation point untergeordnet ist. Erfolgt die Abbildung eines anwenderorientierten Prozessmodells, das mit AOPM oder BPMN erstellt wurde, auf ein generisches Datenmodell, kann damit ein Großteil der Komplexität vor den Anwendern versteckt werden.

Besonders hervorzuheben ist das *Leere Element* (Knoten) des generischen Datenmodells. Damit können zwingende und optionale Varianten in natürlicher Art und Weise modelliert werden. Die bekannten Modellierungssprachen besitzen dazu kein korrespondierendes Element. Die zur Unterscheidung nötigen Informationen müssen dem Kontext entnommen werden, was nicht immer als trivial anzusehen ist und daher eine wesentliche Herausforderung bei der Erstellung des generischen Datenmodells darstellt. Lediglich beim funktionalen Aspekt kann ein fehlender Prozessschritt als leeres Element aus dem Modell herausgelesen werden; bei allen anderen Aspekten, ist dies in der graphischen Darstellung eines Aspektes nicht möglich.

Ähnlich verhält es sich in Bezug auf die *Implikationen* (= Verbindungen). Während in AOPM zwar alternative Kommentare verwendet werden, besteht bei den anderen Modellierungssprachen keine Möglichkeit der Anzeige von Abhängigkeiten zwischen Entscheidungen und deren Alternativen. Implikationen tragen wesentlich dazu bei, den Entscheidungsraum innerhalb eines Prozessmodells zu organisieren und auf ein sinnvolles Maß zu begrenzen. Diese beiden Aspekte unterstreichen nochmals, unabhängig von der Konfigurierbarkeit, die Mächtigkeit und Ausdruckskraft des generischen Datenmodells. Es sollte dabei berücksichtigt werden, dass die Implikationen eine Herausforderung an die Validierung der resultierenden Modelle stellen.

Es müssen jedoch die anwenderorientierten Prozessmodelle auf die spezielle Struktur des generischen Datenmodells abgebildet werden. Wird dieser Schritt manuell ausgeführt, stellt dies einen zeitaufwendigen sowie fehleranfälligen Vorgang dar. Daher ist es empfehlenswert diesen *Transformationsschritt* zu automatisieren. Dann reicht es aus, die Korrektheit des Transformationsprogrammes zu validieren, was weit aus effizienter ist [ReMA06]. In dem Zusammenhang ist zu beachten, dass auch die Validierung des anwenderorientierten Prozessmodells mittels geeigneter Validierungsmechanismen nötig ist [LaVi01] [LiSS94], bevor es für die Konfiguration verwendet wird.

Generisch		UML	AOPM	EPK	BPMN
Knoten	Elementares Element	Aktivität	Elementarer Prozess	Funktion	Activity
		Objekt	Datum	Informations-objekt	Event
		-	Rolle	Organisations-einheit	Data Object
		-	Werkzeug	-	Pool, Lane
	Leeres Element	(Kontext)	(Kontext, Kommentare)	(Kontext)	(Kontext)
	Komposites Element	-	Kompositer Prozess	-	-
		Konjunktion	Konjunktion	Konjunktion	Gateway
	Alternative		Alternative	Alternative	Gateway
	exklusive Alternative	exklusive Alternative	exklusive Alternative	exklusive Alternative	Gateway
	Kanten	-	Schnittstelle des organisatorischen und operationalen Aspektes, Datenfluss, hierarchische Modellierung der Prozesse	Kante zu Informations-objekt	Assoziation
				Kante zu Organisations-einheit	Description
Verbindungen	Flüsse	Kontrollfluss	Kontrollfluss	-	Sequence flow
		Objektfluss	Datenfluss	Informationsfluss	Message flow
	Implikation	(Kontext)	(Kontext, Kommentare)	(Kontext)	(Kontext)

Tabelle 6-4 Zuordnung der Elemente - Sprachunabhängigkeit

Da das generische Datenmodell nicht der sonst gängigen Darstellung von Prozessen entspricht, kann es nicht für die Ansicht durch den Prozessausführenden und/oder – verantwortlichen verwendet werden. Mittels des generischen Datenmodells ist es möglich sehr komplexe und umfassende Variantenräume und deren Strukturen kompakt und präzise zu formulieren.

6.5.6 Das Anwenderorientierte Prozessmodell

Für das anwenderorientierte Prozessmodell wird in dieser Arbeit exemplarisch die aspektorientierte Modellierung verwendet (siehe Kapitel 3.3.1.4). Dies wird damit begründet, dass die erarbeiteten Konzepte auf der Erfahrung der Erstellung von Prozessmodellen aufbauen, die alle mit AOPM erstellt worden sind.

6.5.6.1 Use Case - AOPM

Um dem Anwender die Varianten in einem Prozessmodell aufzeigen zu können, ist in dem AOPM-Modell die Anzeige der *variation points* in Bezug auf die verschiedenen Elemente der Prozessmodelle sinnvoll. Die Anzeige kann prinzipiell als optional angesehen werden, da zur umfassenden Abbildung der Variabilität das generische, konfigurierbare Datenmodell in Kapitel 6.5.5 konzipiert wurde; die Anpassungen lassen sich hauptsächlich aus dem Aspekt der Benutzerfreundlichkeit heraus motivieren.

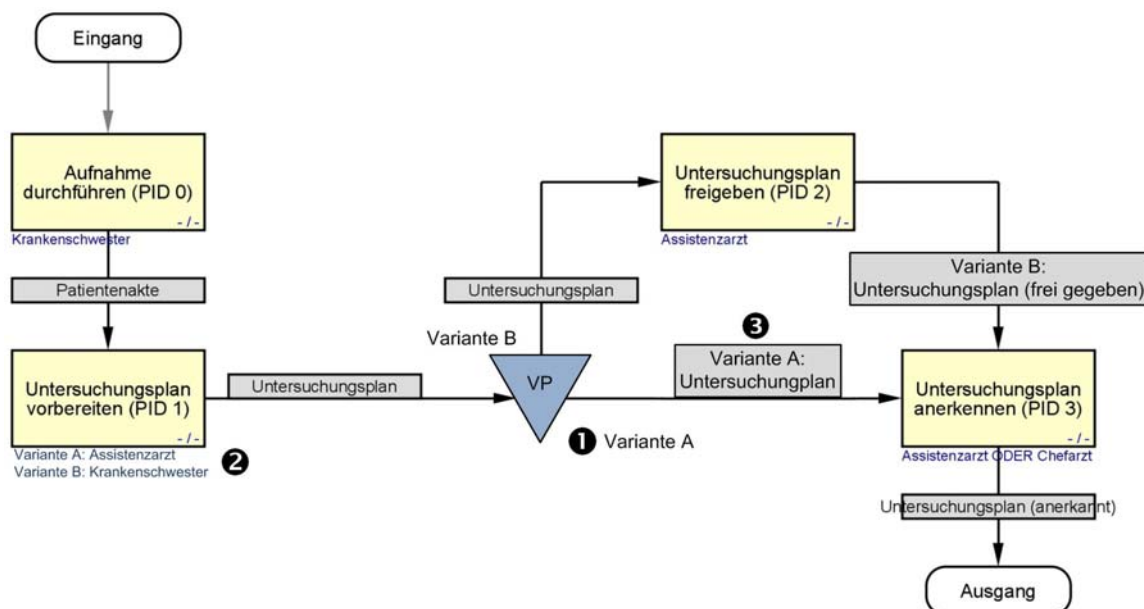






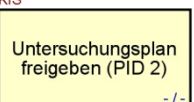



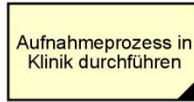


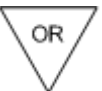


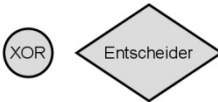


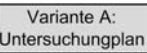
Abbildung 6-18 Konfigurierbares AOPM-Prozessmodell

Folgende Anpassungen in Bezug auf die Darstellung der *variation points* wurden daher vorgenommen:

- Varianten in Bezug auf die Prozesse (*Funktionsorientierte Varianten*) werden graphisch mittels eines expliziten *variation points* in Form eines Dreiecks mit nach unten gerichteter Spitze dargestellt (siehe variation point ❶ in Abbildung 6-18). Die aus diesem Modellierungselement ausgehenden Kontrollflüsse führen zu den *variation options*, d.h. zu weiterführenden Prozessen. Die zu den *variant options* hinführenden Kontrollflüsse sind außerdem mit dem Attribut ‚Variante‘ und einer alphabetischen Indexierung gekennzeichnet. Dies dient der Anzeige, zu welcher Variante die jeweilige *variant option* gehört. Die Beschriftung am Kontrollfluss wurde gewählt, weil es vorkommen kann, dass sich eine Variante durch das Nicht-Vorhandensein eines Prozessschrittes auszeichnet, wie es in dem Beispiel in Abbildung 6-18 der Fall ist. Eine Kennzeichnung eines Prozessschrittes ist dann nicht möglich.
- Bei dem Beispiel an variation point ❶ handelt es sich aufgrund des Auslassens eines Prozessschrittes zugleich um eine *verhaltensorientierte Variante*. Auch bei diesen wird also der dazugehörige *variation point* in Form eines Dreiecks mit nach unten gerichteter Spitze dargestellt. Im Unterschied zur funktionsorientierten Variante müssen pro *variant option* keine unterschiedlichen Prozesse definiert sein. Denn hier kommt es auf die Ablaufreihenfolge der Schritte an.
- *Organisatorische und operationale Varianten* werden *textuell* gekennzeichnet. Bei den Typbezeichnungen dieser beiden Aspekte wird das Attribut „Variante“ sowie eine alphabetische Indexierung vorangestellt (siehe variation point ❷ in Bezug auf den organisatorischen Aspekt). Dazu kann für die beiden Aspekte jeweils ein neues, separates Modellierungskonstrukt spezifiziert werden, oder bei dem bisherigen Modellierungselementen das Attribut ‚Variant‘ bei der Definition optional mit angegeben werden, so dass es bei Nicht-Verwendung nicht mit angezeigt wird.
- Auch für die Darstellung der *datenorientierten Variante* muss kein separates Modellierungselement definiert werden (siehe ❸ in Abbildung 6-18). Wie beim organisatorischen bzw. operationalen Aspekt erfolgt eine Anzeige in *textueller* Form durch das zusätzliche Attribut ‚Variante‘.

6.5.6.2 Bewertung

Bei dem anwenderorientierten Prozessmodell wird eine dem Anwender vertraute Darstellung verwendet. Tabelle 6-5 zeigt die konzeptionelle Abbildung des anwenderorientierten Modells nach AOPM auf das generische Datenmodell. Durch die primär textuelle Anzeige der Variabilität über Attribute muss lediglich für den funktionsorientierten Aspekt ein neues Modellierungskonstrukt eingeführt werden. Die Änderungen fallen somit sehr gering aus. Dies wirkt sich positiv aus auf die Benutzerfreundlichkeit der Modellierungssprache, wie auch auf den Aufwand der implementierungstechnischen Änderungen.

Elemente		Generisch	AOPM	
Knoten	Elementares Element		Elementarer Prozess	
		Output: Patientenakte	Datum	
		Input: Patientenakte		
		Rolle: Krankenschwester	Rolle	
		Werkzeug: KIS	Werkzeug	
	Leeres Element		(Kontext, Kommentare)	(Kontext) 
	Komposites Element		Kompositer Prozess	
			Konjunktion	
	Alternative		Alternative	
	exklusive Alternative		exklusive Alternative	
	Variation point		Prozesse/ Kontrollfluss	
			Datum	

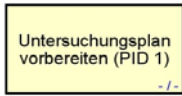
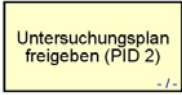






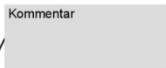
			Rolle	 Variante A: Assistenzarzt Variante B: Krankenschwester
			Werkzeug	Variante A: KIS Variante B: Karteikarte 
Kanten	Kanten		Schnittstellen der Aspekte, hierarchische Modellierung der Prozesse	-
	Flüsse		Kontrollfluss	
Verbindungen			Datenfluss	
	Implikation		(Kommentare, Kontext)	(Kontext) 

Tabelle 6-5 Abbildung generisches Datenmodell - AOPM

Doch weist diese Art der Modellierung einige Schwachstellen auf: Es ist nicht möglich die Variabilität in den Prozessen vollständig, kompakt und zugleich übersichtlich darzustellen. Die Darstellung der variation points erfolgt nicht einheitlich, sondern gemischt in textueller und graphischer Form.

Die Zusammengehörigkeit zwischen den einzelnen variation points ist im Kontext einer Variante nur in Bezug auf ein Prozessblatt leicht zu identifizieren. Bei prozessblattübergreifender Zusammengehörigkeit, sei es auf derselben Ebene oder aufgrund kompositer Prozesse über verschiedene Ebenen des Prozessmodells verteilt, fehlt es an Übersichtlichkeit sowie Eindeutigkeit. Die fehlerfreie Erkennung oder das Ableiten einer einzelnen Variante gestaltet sich somit aufwändig.

6.5.7 Evaluation des Konfigurationskonzeptes

Zur Evaluierung des Konfigurationsansatzes und dem dabei prinzipiell durchzuführenden Zuordnung der Elemente des anwenderorientierten Prozessmodells zu den Elementen des generischen Datenmodells wurden unter Verwendung der Prozessmodelle aus den im Kontext dieser Arbeit relevanten Lehrstuhlprojekten (siehe Kapitel 4) Fallstudien durchgeführt. Ziel war es herauszufinden, ob die vorhandenen anwenderorientierten Prozessmodelle in das entwickelte Datenmodell übertragen, daraufhin konfiguriert und das Ergebnis schließlich wieder in der anwenderorientierten Darstellungsform präsentieren werden kann.

Vorgehensweise und Erfahrungen werden in den folgenden Abschnitten beschrieben. Da es um die Evaluation des Konfigurationskonzeptes sowie des generischen Datenmodells als Artefakte an sich ging [HMPR04], wurde nur ein einstufiger Konfigurationsprozess durchgeführt. Zum Zeitpunkt der Evaluation war noch keine Implementierung des Modellierungs- und Konfigurationssystems vorhanden, so dass im Vergleich zur der in Kapitel 6.5.2 vorgestellten Vorgehensweise einzelne Schritte hinzugefügt oder leicht abgeändert wurden. Zudem wurde die Evaluation nicht abschließend, sondern als ein Zwischenschritt im Rahmen der vorliegenden Forschungsarbeit durchgeführt. Erfahrungen und Ergebnisse aus der hier vorzustellenden Evaluation sind somit bereits in Teilen bzw. soweit wie möglich in dem in Kapitel 6.5.5 vorgestellte generische Datenmodell berücksichtigt.

6.5.7.1 Vorgehensweise

In diesem Abschnitt werden die durchgeführten Schritte der Evaluation, angefangen mit der Analyse der vorhandenen Prozessmodelle bis hin zur Präsentation der Konfigurationsergebnisse, beschrieben.

Erstellung der integrierten anwenderorientierten Prozessmodelle

In den aus den Lehrstuhlprojekten vorhandenen AOPM-Prozessmodellen galt es zunächst die Varianten, d.h. die variation points und variant options, zu identifizieren und zu definieren. Anschließend erfolgte die Integration der einzelnen Varianten in ein gemeinsames Modell, wenn die Varianten nicht schon per se derart modelliert worden waren.

Aus jedem der beiden Projekte wurden drei Anwendungsfälle, d.h. jeweils drei Gruppen von Varianten, ausgewählt. Es sollte darauf hingewiesen werden, dass in den Prozessmodellen der jeweiligen Projekte noch weitaus mehr Varianten zu finden sind. Für die Evaluation wurde zur Begrenzung des Aufwands nur eine Auswahl verwendet. Insgesamt wurden daraus sechs integrierte Prozessmodelle erstellt.

Aus dem universitätsbezogenen Prozessmodell, dem „Studenten-Lebenszyklus“, wurden folgenden drei Gruppen ausgewählt:

- (1) *Einführung eines neuen Studiengangs vorbereiten*: Dieser Prozess zeigt die auf den verschiedenen hierarchischen Ebenen der Universität notwendigen Schritte für die Beschlussfassung zur Einführung eines Studiengangs.
- (2) *Bewerbung durchführen*: Dieser Prozess stellt dar, wie sich Studenten (national/international, Degree/ Non-Degree, ...) an der Universität bewerben müssen, wenn der betreffende Studiengang nicht zulassungsfrei ist.
- (3) *Prüfungsverfahren durchführen*: Hier wird die Organisation sowie Durchführung der Prüfungen (schriftlich oder mündlich) an einem Lehrstuhl dargestellt.

Aus dem medizinischen Prozessmodell wurden folgende drei Gruppen ausgewählt:

- (1) *Aufnahme eines Patienten durchführen*: Dieser Prozess wird von einem Arzt bei der Einlieferung eines Patienten in die Klinik durchgeführt.
- (2) *Entlassung durchführen*: Dieser Prozess wird durchgeführt, um den Klinikaufenthalt eines Patienten offiziell zu beenden, so dass dieser die Klinik verlassen kann.
- (3) *Rechnung schreiben*: Dieser Prozess enthält alle Schritte, die notwendig sind, um die im Rahmen eines Krankenhausaufenthalts durchgeführten Behandlungen aufzulisten und in Rechnung zu stellen.

Zu Anwendungsfall (3) ist in Abbildung 6-20 das integrierte Prozessmodell zu sehen, ohne dass die variation points explizit modelliert sind (das Modellierungswerkzeug ermöglichte dies zum Zeitpunkt der Evaluation nicht). In Abbildung 6-21 und Abbildung 6-19 sind die Prozessmodelle der beiden identifizierten Varianten separat dargestellt.

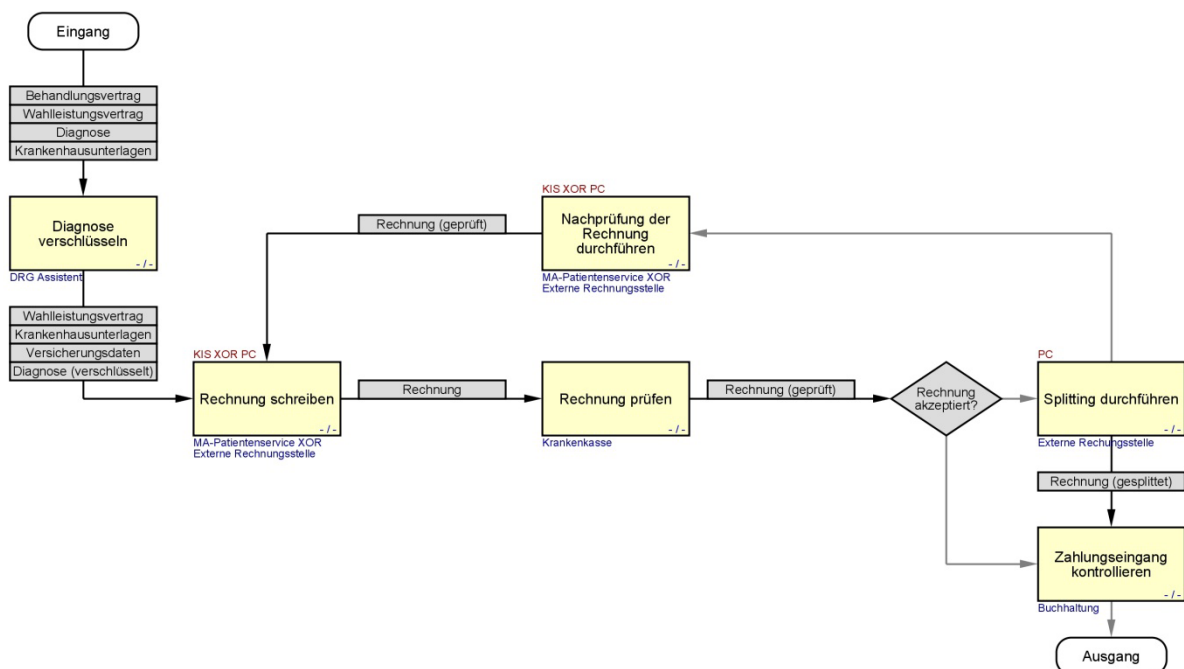


Abbildung 6-19 Rechnung schreiben – integriertes AOPM-Prozessmodell

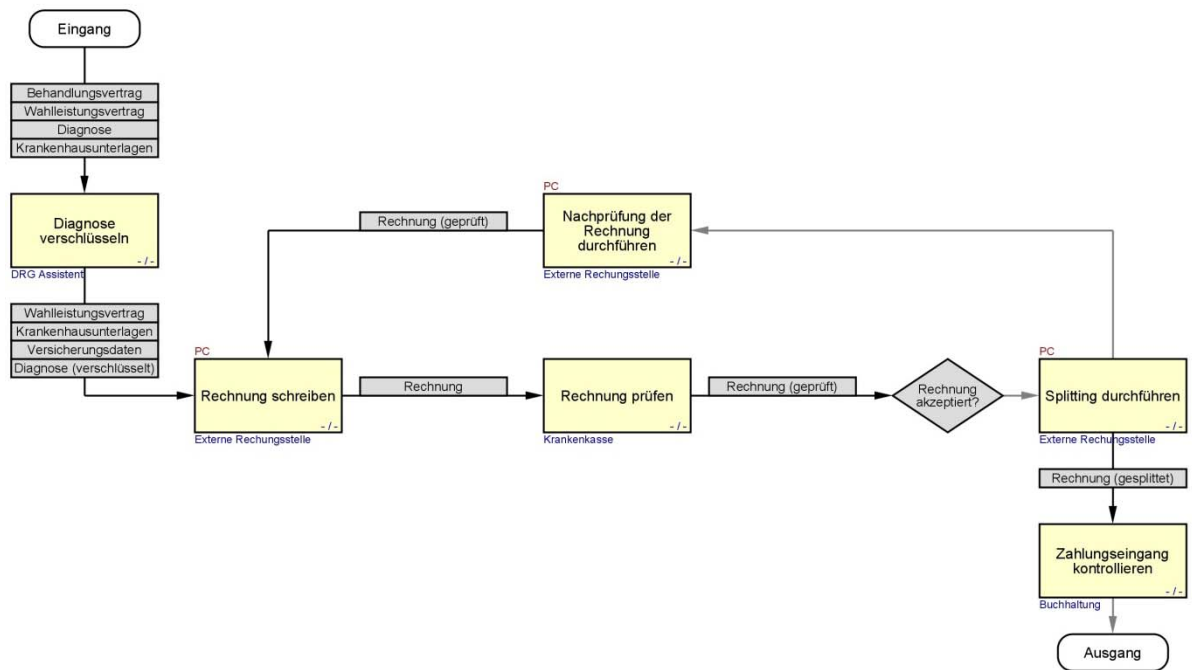


Abbildung 6-20 Rechnung schreiben - Variante 1

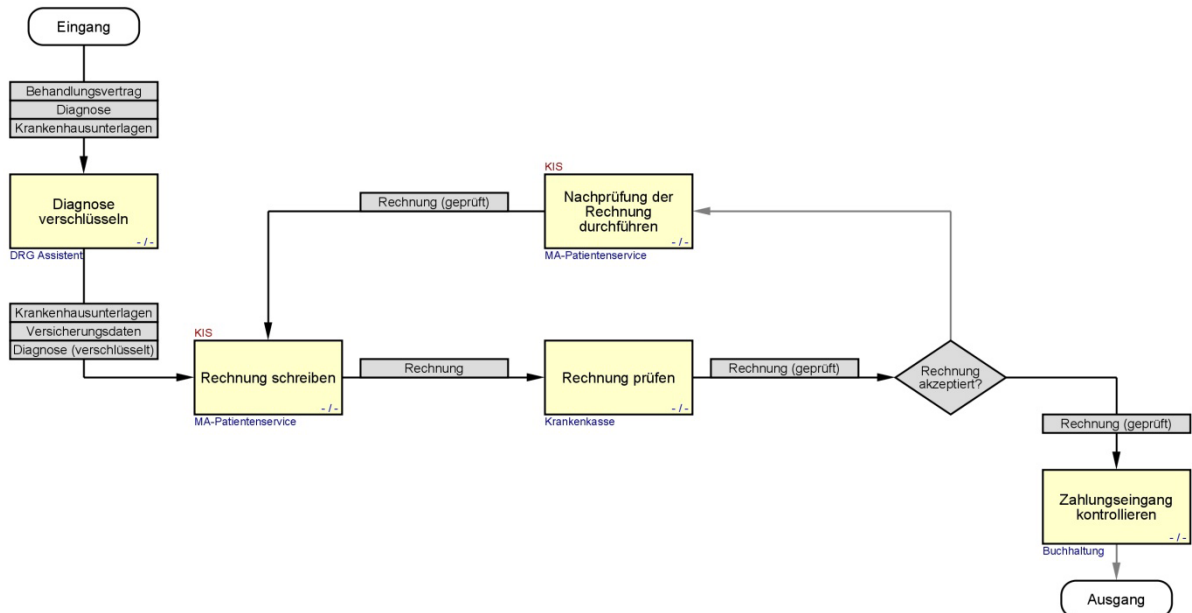


Abbildung 6-21 Rechnung schreiben – Variante 2

Transformation

Im nächsten Schritt galt es die integrierten AOPM-Prozessmodelle in das generische, konfigurierbare Datenmodell zu übertragen; das Ergebnis für Anwendungsfall (3) „Rechnung schreiben“ ist in Abbildung 6-22 zu sehen. Da zum Zeitpunkt der Evaluation keine Implementierung des Ansatzes in einem Modellierungswerkzeug existierte, wurde dieser Schritt manuell durchgeführt. Damit war es jedoch leicht möglich, Informationen, die informell aus den Interviews der Prozessaufnahme oder im AOPM-Modell nur textuell als Kommentar vorhanden waren, in das generische Datenmodell zu integrieren. Dies bezog sich meistens auf die Definition von Abhängigkeiten zwischen den Prozesselementen oder optionalen Entscheidungen.

Vorbereitung der Konfiguration

Da, wie bereits erwähnt, zum Zeitpunkt der Evaluation noch keine Implementierung des Prozesskonfigurators vorhanden war, wurde zur Erleichterung des weiteren Vorgehens für jeden Anwendungsfall eine sog. *variant list* definiert. Diese dokumentiert bei jedem Anwendungsfall für die jeweils dazugehörige Variante

- *welche variation points für eine konkrete Variante relevant sind*
Für das Prozessmodell in Abbildung 6-19 sind zum Beispiel drei datenorientierte und zwei funktionale variation points relevant.
- *welche variant option der ausgewählten variation points zu einer Variante gehört*
Für das Beispiel in Abbildung 6-19 wurde zum Beispiel an dem ersten datenorientierten variation point (Input) für Variante (3a) der „Wahlleistungsvertrag“ als variant option definiert, für Variante (3b) ist kein Datum definiert. Abhängigkeiten zwischen variant options wurden dabei ebenfalls spezifiziert.

Konfiguration

Für jede Variante wurden die relevanten variation points gemäß der vorab definierten *variant list* aufgelöst. Diese erfolgt durch Löschen der nicht zu der jeweiligen Variante gehörenden variant options. Da es sich um einen einstufigen Konfigurationsprozess handelte, durfte nach der Konfiguration pro variation points nur noch eine variant options vorhanden sein; den variation point galt es jeweils durch diese variant option, d.h. ein aspektororientiertes Prozesselement, zu ersetzen. Daran schloss sich die Überprüfung der Korrektheit des resultierenden generischen Datenmodells an. Anschließend erfolgte dessen Retransformation in die AOPM-Notation. Alle Teilschritte wurden aufgrund fehlender Werkzeugunterstützung wiederum manuell ausgeführt.

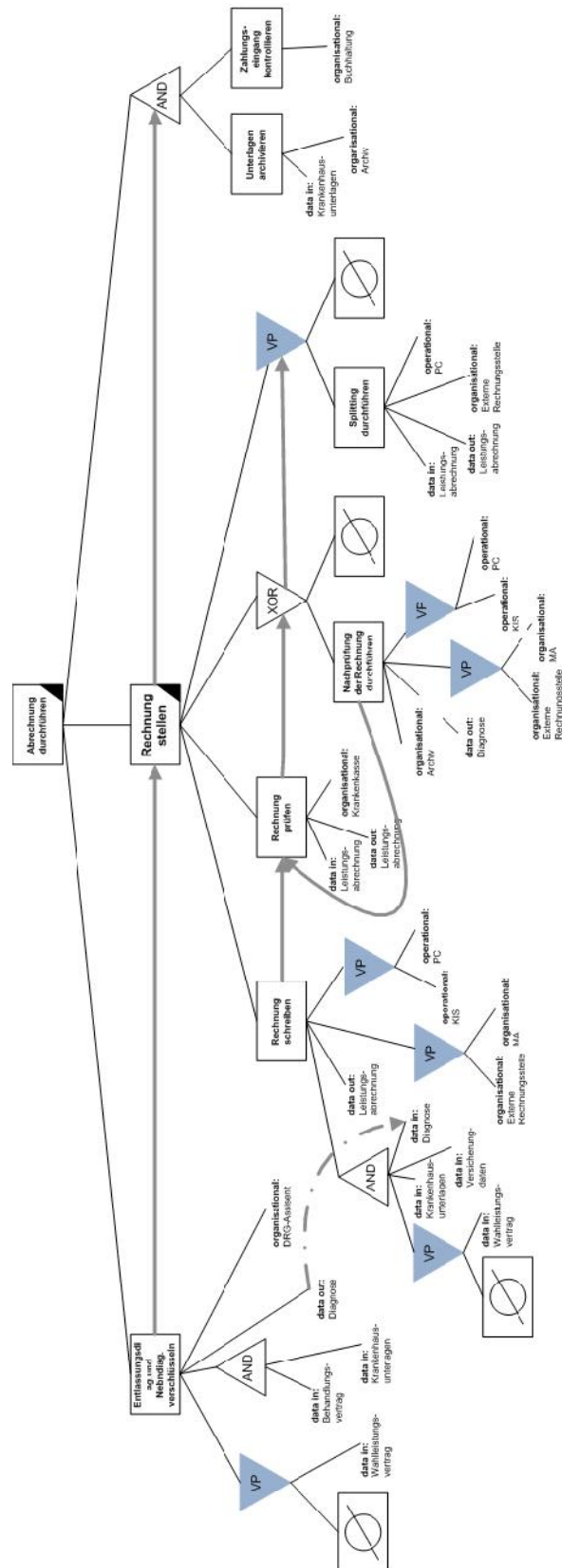


Abbildung 6-22 Rechnung schreiben – generisches Datenmodell

6.5.7.2 Beobachtungen

Allgemein kann festgehalten werden, dass für alle Gruppen von Varianten sowohl integrierte Prozessmodelle mit AOPM als auch als generische Datenmodelle erstellt werden konnten. Auf Basis der konzeptionellen Definition der einzelnen Varianten (Vorbereitung der Konfiguration mit Hilfe der *variant list*) konnten diese auch wieder aus den integrierten, generischen Datenmodellen abgeleitet und in die AOPM-Modellierungssprache zurück übertragen werden. Dies zeigt, dass das Konzept sowohl die Integration von Varianten in einem Modell, als auch das Ableiten einzelner Varianten aus einem Gesamtmodell ermöglicht. Die wesentliche Ziele bzw. Anforderungen sind somit erfüllt.

Erstellung der integrierten AOPM-Prozessmodelle

Die Identifikation der Varianten in den vorhandenen AOPM-Prozessmodellen an sich war mit wenig Aufwand verbunden, nicht jedoch deren Integration in ein gemeinsames Modell.

Der Abgleich scheinbar übereinstimmender Prozesselemente (insbesondere von Prozessschritten sowie Datenelementen) war oftmals nur nach reichlicher Überlegung und vielen Diskussionen möglich. Diese waren vor allem notwendig aufgrund divergierender qualitativer Bedeutungen verschiedener Prozesselemente (vor allem bei den Datenelementen) oder unterschiedlicher Beziehungen zu verschiedenen anderen Prozesselementen. Es galt, den Verlust wichtiger Informationen im Bezug auf die Prozessabläufe zu vermeiden.

Außerdem musste darauf geachtet werden, keine Prozesselemente zu vergessen, keine existierenden Beziehungen zu übersehen (dies galt vor allem bei der Definition des Kontrollflusses) oder aber falsche Verbindungen zu modellieren. Schließlich musste darauf geachtet werden, dass das integrierte AOPM-Prozessmodell prinzipiell ausführ- bzw. umsetzbar ist, d.h. korrekt definiert war [LaVi01] [LiSS94] [vADG10]. Da dieser Validierungsschritt manuell durchgeführt wurde, sind die Ergebnisse mit einem gewissen Unsicherheitsfaktor zu bewerten. Ein hohes Maß an Modellierungserfahrung war hier essentiell.

Transformation

Der ebenfalls manuell durchgeführte Transfer der AOPM-Prozessmodelle zum generischen Datenmodell war vor allem zeitaufwändig und die Gefahr von Fehlern sehr hoch. Eine automatisierte Lösung ist daher wünschens- bzw. erstrebenswert. Sie stand zum Zeitpunkt der Durchführung der Studie Mitte 2010 jedoch noch nicht zur Verfügung. In diesem Fall wäre die Validierung der Korrektheit des Transformationsprogrammes ausreichend, statt das erzeugte generische Datenmodell auf Korrektheit an sich und im Hinblick auf das abzubildende AOPM-Prozessmodell überprüfen zu müssen. Dieses Vorgehen ist weitaus effizienter.

Aufgrund dessen, dass das generische Datenmodell als Graph konzipiert ist, galt es während der Transformation einige Herausforderungen zu bewältigen. Dies galt primär für die Abbildung des *Kontrollflusses*, vor allem wenn dieser Konnektoren oder Schleifen enthielt. Es galt darauf zu achten, keine Kontrollflusselemente zu vergessen, da dies sonst zu sog. „dead ends“ im Prozessablauf führen könnten. Durch das Durchspielen der einzelnen möglichen Abläufe gemäß den zugrundegelegten AOPM-Prozessmodellen wurde versucht irrtümlich definierte oder fehlende Verbindungen aufzudecken und zu korrigieren.

Für das Modellierungselement *leeres Element* im generischen Datenmodell gibt es in einem AOPM-Modell kein korrespondierendes Element (Tabelle 6-4). Somit musste dieses zur Modellierung in dem generischen Datenmodell aus dem Kontext der Prozesse der AOPM-Modelle abgeleitet werden, was zum Teil nur mit entsprechenden Hintergrundinformationen möglich war. Dabei besteht bei der Verwendung von i>pm als Modellierungswerkzeug die Möglichkeit, dass diese Informationen in einer wenn auch nicht vollständig formalen Art und Weise über Kommentare und somit in textueller Form im Modell beschrieben werden können. Der Transformationsschritt war ein sehr herausfordernder sowie kritischer Schritt. Die manuelle Integration dieser Elemente war somit sehr zeitaufwändig und, aufgrund der Gefahr von Missverständnissen, sehr fehleranfällig. Ein hohes Maß an Modellierungserfahrung war auch hier essentiell. Implizites, auf Erfahrung basierendes Domänen- bzw. Fachwissen der Anwender war notwendig, welches dem eigentlichen Modellierer oftmals nicht zur Verfügung steht.

Soeben Gesagtes gilt auch für die Spezifikation der *Abhängigkeiten* bzw. *Implikationen*, da sich auch diese nicht explizit im AOPM-Prozessmodell modellieren lassen. Das generische Datenmodell wurde sorgfältig mit den Aufzeichnungen zu den Interviews der Prozessanalyse abgeglichen, um eine inkorrekte Modellierung der Sachverhalte zu verhindern.

Vorbereitung der Konfiguration

Wie bereits erwähnt, war die Identifikation der Varianten wenig problematisch und somit auch die Definition der variation points und variant options in der *variant list*. Es handelte sich lediglich um einen Dokumentationsschritt.

Konfiguration

Da die eigentliche Konfiguration an sich manuell ausgeführt wurde, war die korrekte Individualisierung der generischen Datenmodelle ebenfalls eine Herausforderung. Dies traf ebenso für die Retransformation der resultierenden generischen Datenmodelle in die POPM-Modellierungssprache zu. Zur Qualitätssicherung wurde die Validierung der finalen POPM-Prozessmodelle in mehreren Zyklen durchgeführt wurde, wodurch der Zeitaufwand stieg.

Fazit

Trotz einiger (aktueller) Problemfelder scheint das entwickelte Konzept die Anforderungen zu erfüllen. Weiterer Forschungsbedarf besteht im Hinblick auf die derzeit noch manuell in das generische Datenmodell nachgetragenen Kontextinformationen, insbesondere Abhängigkeiten (siehe dazu zum Beispiel [HaBR08a] oder [LaRo08a]). Sie werden meist über Implikationen oder leere Elemente in dem generischen Datenmodell abgebildet, doch gibt es in dem anwenderorientierten AOPM-Prozessmodell bisher nur die unzureichende Darstellung über Kommentare. Wird dieses Modellierungsproblem nicht formal gelöst, so kann es auch bei der zu entwickelnden Werkzeugunterstützung nicht mit einbezogen werden.

Darüber hinaus ist für die Weiterentwicklung des Ansatzes im Hinblick auf die Modellierung eine entsprechende Werkzeugunterstützung essentiell. Darüber hinaus ist eine direkte Kopplung der beiden Modelle zu definieren sowie das Konfigurationsprinzip der gewünschten Prozessvariante zu entwickeln.

In ihrer Kombination ist mit diesen beiden Modellen trotz der genannten Schwachstellen prinzipiell eine umfassende Modellierung sowie Konfiguration von Prozessvarianten möglich. Die Dokumentation der Variabilität kann in vollständigem Umfang in dem generischen Datenmodell und eine benutzerfreundliche Anzeige in Form des Anwender-orientierten Prozessmodells erfolgen.

6.5.8 Prototypische Implementierung der Datenbasis in OMME

Für eine prototypische Implementierung einer Modellierungsumgebung wurde die *Open Meta Modeling Environment OMME* [Volz11] verwendet, die bereits in Kapitel 5.6 eingeführt worden ist. Das generische Datenmodell gilt es gemäß des in Kapitel 6.5.5 beschriebenen Konzeptes zu integrieren. Die in Form eines Use Case in OMME integrierte AOPM-Modellierungssprache wird exemplarisch für die anwenderorientierte Darstellung weiter- bzw. wiederverwendet. Ein Beispiel aus der aktuellen Ansicht ist in Abbildung 6-23 zu sehen. Im oberen (rechten) Bereich der Abbildung ist der Ausschnitt eines generischen Datenmodells zu sehen, in unteren (rechten) Bereich der Teilbereich des entsprechenden AOPM-Modells.

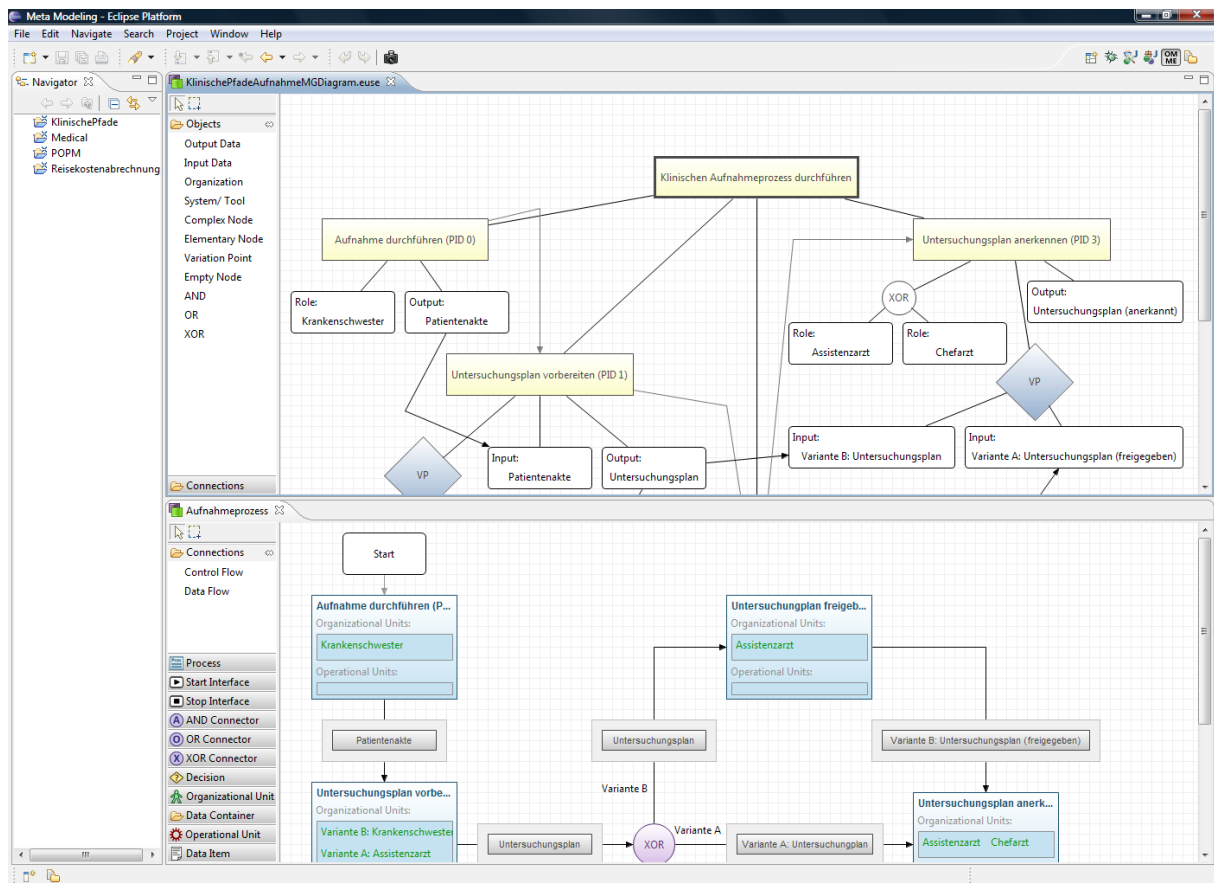


Abbildung 6-23 Screenhoot OMME

6.5.8.1 Das generische Datenmodell in OMME

Das generische Datenmodell besteht in OMME lediglich aus den Ebenen M2 und M1. Auf M2 wird die Modellierungssprache zur Erstellung des Modells bzw. des Graphen deklariert, auf M1 erfolgt die eigentliche Modellierung.

```

1  concept Graph {
2      attributes {
3
4      string name;
5
6      concept Node nodes { multiplicity = zeroOrMore; }
7      concept Edge edges { multiplicity = zeroOrMore; }
8      concept Link links { multiplicity = zeroOrMore; }
9  }
10 }
```

Abbildung 6-24 Deklaration des Graphen auf M2

Zur Definition der Modellierungssprache auf M2 müssen der *Graph* und seine wesentlichen Bestandteile definiert werden (siehe Abbildung 6-24). Dies erfolgt in Form von Knoten, Kanten und Verbindungen, wie es in Abbildung 6-24 in den Zeilen 6 - 8 zu sehen ist. Was in den dazugehörigen geschweiften Klammern steht, ist für den aktuellen Kontext nicht relevant. Außerdem kann dem Graphen ein Name gegeben werden (Zeile 4).

Die Knoten, Kanten und Verbindungen müssen nun noch weiter bzgl. ihrer Bedeutung und ihrer Beziehung zueinander konkretisiert werden, was ebenfalls auf Ebene M2 erfolgt.

```

1  abstract concept Node {
2      attributes {
3          concept Edge inboundEdges { multiplicity = zeroOrMore; opposite = target; }
4          concept Edge outboundEdges { multiplicity = zeroOrMore; opposite = source; }
5
6          concept Link inboundLinks { multiplicity = zeroOrMore; opposite = target; }
7          concept Link outboundLinks { multiplicity = zeroOrMore; opposite = source; }
8      }
9  }
10
11 package Nodes {
12     concept ComplexNode extends Node { attributes {
13         string name;
14         string variants {multiplicity = zeroOrMore;} } }
15
16     concept ElementaryNode extends Node { attributes {
17         string name;
18         string variants {multiplicity = zeroOrMore;} } }
19
20     concept VariationPoint extends Node { attributes { string name; } }
21
22     concept AND extends Node {}
23     concept OR extends Node {}
24     concept XOR extends Node {}
25
26     concept EmptyNode extends Node {}
27
28     concept OutputData extends ElementaryNode {}
29     concept InputData extends ElementaryNode {}
30     concept Operation extends ElementaryNode {}
31     concept Organization extends ElementaryNode {}
32 }

```

Abbildung 6-25 Deklaration der Knoten auf M2

Bei den *Knoten* (= *Node*) wird festgelegt, dass diese sowohl ein- als auch ausgehende Kanten und Verbindungen haben können. Dies erfolgt in Abbildung 6-25 in den Zeilen 1 - 7.

Es gilt darüber hinaus in den Zeilen 11 - 23 die einzelnen Knotentypen in Bezug auf die bei AOPM definierten Prozesselemente zu spezifizieren, wie zum Beispiel mit *ComplexNode* in Zeile 12 der kompositen Prozess. Wenn erwünscht können die Prozesselemente mit Attributen näher beschrieben werden, wie es bei *ComplexNode* mit **string** *name*, zu sehen ist. Dem kompositen Prozess kann damit ein Name gegeben werden. Der elementare Knoten wird nach demselben Prinzip definiert.

Über das Attribut `string variants` für den kompositen Knoten (Zeile 14) bzw. für den elementaren Knoten (Zeile 18), wird die Variantenzugehörigkeit, das heißt die Bezeichnung der Variante, an den Knoten bzw. Prozesselementen angegeben. Dieses Attribut wird nur dann genutzt, wenn das Prozesselement einem variation point zugeordnet ist. In diesem Fall ist das Prozesselement dann zugleich als variant options aufzufassen.

Ein variation point enthält nur das Attribut für eine mögliche Bezeichnung (Zeile 20).

Die Konnektoren (Zeile 22 - 24) werden ganz ohne Attribute definiert. Selbiges gilt für das leere Element (Zeile 26).

Die Modellierungselemente für den Daten-orientierten, operationalen sowie organisatorischen Aspekt (Zeile 24 - 27) werden von dem in Zeile 14 definierten elementaren Knoten abgeleitet. Durch das Ableiten übernehmen sie automatisch die für den elementaren Knoten definierten Attribute.

```

1  concept Edge {
2      attributes {
3          string tag;
4      }
5      concept Node source { opposite = outboundEdges; }
6      concept Node target { opposite = inboundEdges; }
7  }
8  }
```

Abbildung 6-26 Deklaration der Kanten auf M2

```

1  abstract concept Link {
2      attributes {
3          string tag;
4      }
5      concept Node source { opposite = outboundLinks; }
6      concept Node target { opposite = inboundLinks; }
7  }
8  }
9
10 package Links {
11     concept ControlFlow extends Link {}
12     concept DataFlow extends Link {}
13     concept Implication extends Link {}
14 }
```

Abbildung 6-27 Deklaration der Verbindungen auf M2

Die *Kanten* (= *Edge*) werden als Verbindungselemente zwischen Knoten definiert (siehe Abbildung 6-26), wobei einer als Start-, der andere als Zielknoten bzw. einer als Ober- und einer als Unterteil angesehen wird (Zeile 5 - 6). Damit sind die Kanten zwar gerichtet definiert. Dies wird aber nicht in der graphischen Darstellung der späteren Modelle umgesetzt (siehe dazu in Abbildung 6-13 zum Beispiel die Kanten „PID 0“ zu „Organisation: Krankenschwester“).

Die *Verbindungen* (= *Links*) werden ebenfalls als Verbindungselement zwischen einem Start- und einem Zielknoten definiert (siehe Abbildung 6-27, Zeile 5-6). Die damit gerichtet definierten Verbindungen werden als solche auch so weiterverwendet. Dies spiegelt sich auch in den Arten von Verbindung wieder, die zusätzlich in den Zeilen 11 - 13 spezifiziert werden. Es handelt sich um Kontroll- und Datenflüsse sowie die Implikationen. In der graphischen Darstellung werden zudem Pfeile verwendet, wie es in Abbildung 6-28 zu sehen ist.

Damit sind alle wesentlichen Bestandteile des generischen Datenmodells definiert. Die Speicherung erfolgt auch hier, wie für AOPM, in dem *linguistischen Meta Modell* (= LLM). Schließlich muss wieder eine Abbildung auf ein graphisches Modell erfolgen, damit der graphische Editor für die eigentliche Modellierung generiert werden kann. Ein Modellierungsbeispiel ist in Abbildung 6-28 zu sehen.

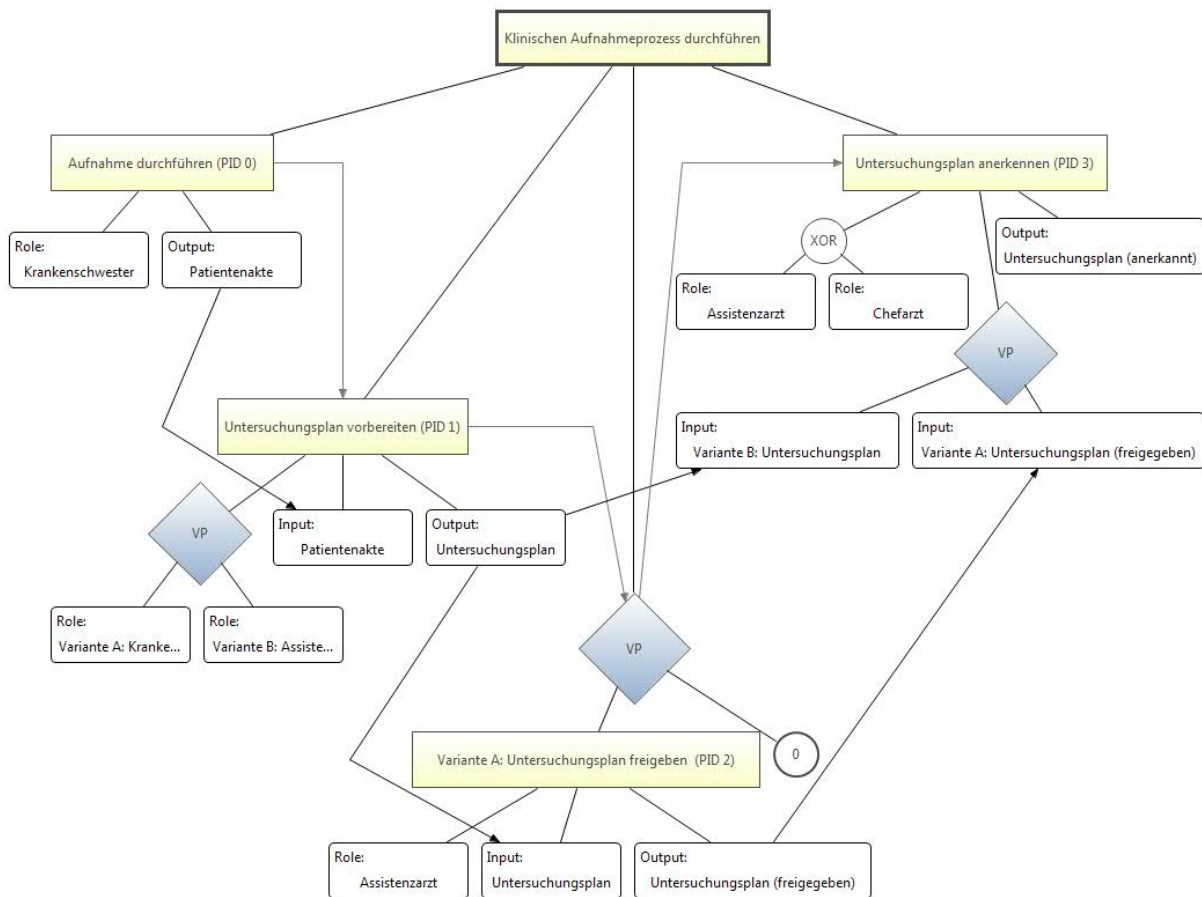


Abbildung 6-28 Screenshot eines generisches Datenmodells in OMME

Die graphische Darstellung ist derzeit noch mit implementierungstechnischen Problemen konfrontiert. Daher werden einige Modellierungselemente, insbesondere die variation points, Konnektoren und das leere Element, anders dargestellt, als es zuvor in 6.5.4 konzeptionell entworfen worden ist. Bei den verschiedenen Kanten und Verbindungen kann lediglich in der

Farbe, nicht jedoch in die Strichart (zum Beispiel gestrichelt und gepunktet) unterschieden werden. Dies wirkt sich negativ auf die Lesbarkeit des generischen Modells aus. Auf die bei AOPM übliche gefärbte Ecke zur Anzeige eines kompositen Prozesses muss ebenfalls verzichtet werden. Stattdessen wurde die Umrandung der Elemente mit einer höheren Strichstärke eingestellt. Das Verstecken der kompositen Elemente in der graphischen Ansicht wäre sinnvoll, ist jedoch aktuell nicht möglich. Zudem kann es dann Probleme bei der Darstellung der Abhängigkeiten geben.

Für weitere technische Aspekte bei der Implementierung in OMME sei auf [Volz11] verwiesen.

6.5.8.2 Das konfigurierbare AOPM-Modell in OMME

In der existierenden AOPM-Modellierungssprache gilt es explizit den funktionalen *variation point* sowie die *varianten-kennzeichnenden Attribute* zu definieren.

```

1  abstract concept variation point instanceof NodeKind {
2      values {
3          boolean supportsInboundControlFlows    = true;
4          boolean supportsOutboundControlFlows    = true;
5      }
6  }
```

Abbildung 6-29 Deklaration des variation points auf M2

Für den funktionalen *variation point* (siehe Abbildung 6-29) muss das linguistische Meta Modell (= LLM) um dieses neue Modellierungselement erweitert werden. Dies erfolgt auf M2, im abstrakten Modellteil APMM, so dass die nun folgende Spezifikation nicht nur für AOPM gültig ist, sondern für jede andere Modellierungssprache, die zusätzlich in OMME definiert wird, wiederverwendet werden kann.

NodeKind in Zeile 1 der Abbildung 6-29, stellt, vereinfacht formuliert, das Basiselement von OMME dar, das der Definition von Modellierungselementen dient. Der variation point ist mit ein- und ausgehenden Kontrollflüssen zu modellieren.

```

1  concept OrganisationalPerspective extends Perspective diCounter 2 {
2      attributes {
3          string displayName { multiplicity = one; }
4          ...
5      }
6  }
```

Abbildung 6-30 Deklaration der organisatorischen variant option auf M3

Bei den nicht funktionalen Aspekten wird die Variabilität über die *Markierung der Prozesselemente* als *variant options*, mit der die direkte Zuordnung zu einer Variante erfolgt, verdeutlicht. Dies erfolgt in der graphischen Darstellung durch eine textuell Angabe.

Konzeptuell wäre diese, wie die Deklaration der Prozesselemente an sich sowie die des funktionalen variation points, auf M2 anzugeben. Aus technischen Gründen (siehe dazu [Volz11]) muss die Spezifikation jedoch auf M3 im APM²M-Modell erfolgen. Bei dem organisatorischen, operationalen und Daten-orientierten Aspekt wird aktuell mit der Bezeichnung des jeweiligen Prozesselements, spezifiziert über `string displayName`, auch die Variantenzugehörigkeit definiert (siehe Abbildung 6-30).

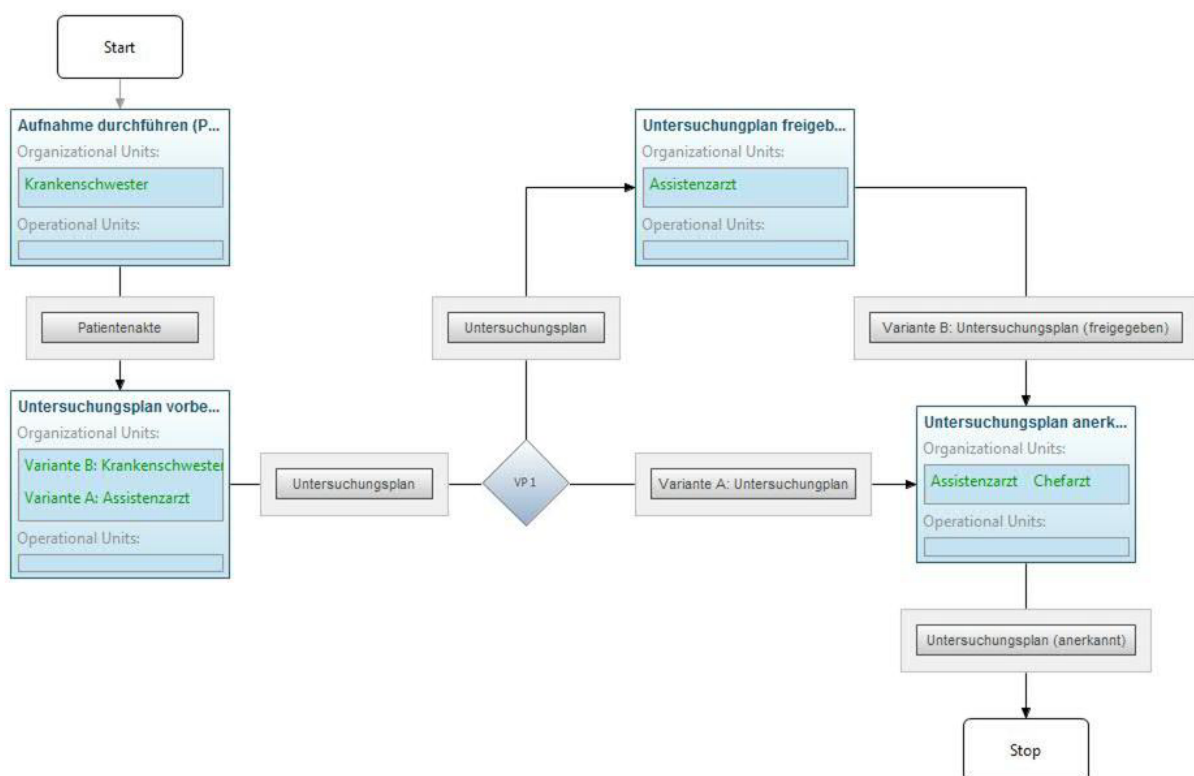


Abbildung 6-31 Konfigurierbares AOPM-Modell in OMME

Damit sind die notwendigen Änderungen an dem Sprach- bzw. Sprachdefinitionsmodell auf M2 bzw. M3 vorgenommen worden. Diese Definitionen können dann mittels der Übertragung in ein graphisches Modell im graphischen Editor zu Modellierung von konfigurierbaren Modellen verwendet werden. Der Ausschnitt eines Beispiels aus dem medizinischen Bereich ist in Abbildung 6-31 zu sehen. Der funktionale variation point kann aufgrund derzeitiger, implementierungstechnischer Konventionen nicht als ein auf der Spitze stehendes Dreieck dargestellt werden, stattdessen wird daher eine Raute verwendet. Im Unterschied zu den bisherigen Prozessabbildungen in dieser Arbeit, wird bei der Verwendung von OMME als

Modellierungswerkzeug (statt i>pm) der organisatorische wie auch operationale Aspekt innerhalb der Prozessschritte angezeigt.

Anstelle der hier vorgenommenen Erweiterung der AOPM-Modellierungssprache kann prinzipiell jede andere beliebige konfigurierbare Prozessmodellierungssprache implementiert werden. Der Ansatz von OMME ist somit sehr generisch und flexibel.

6.6 Fazit

Mit dem vorgestellten *Prozesskonfigurator* wurde ein Gesamtkonzept zum Management variantenreicher Prozesse eingeführt. Ein wesentlicher Beitrag dazu war zum einen das generische, konfigurierbare Datenmodell, das als eine essentiell notwendige, ausdrucksstarke Modellierungsnotation für variantenreiche Prozesse konzipiert worden ist. Zum anderen wurde der gestufte Konfigurationsprozess definiert, der den Anwender mit maximaler Handlungsfreiheit durch den Entscheidungsprozess über eine Variante leitet. Damit ist nicht nur eine methodische, sondern bei vollständiger Implementierung auch eine werkzeugtechnische Unterstützung bei der Konfiguration variantenreicher Prozessmodelle möglich.

Das *generische, konfigurierbare Datenmodell* bildet einen wesentlichen Bestandteil des Prozesskonfigurators. Damit können komplexe Zusammenhänge kompakt, strukturiert und abstrahiert von einer Prozessmodellierungssprache, wie AOPM oder BPMN, dargestellt werden. Die Möglichkeit zur individuellen Gestaltung eröffnet ein breites Einsatzgebiet für den Prozesskonfigurator. Dies wird realisiert durch die Spezifikation der Inhalte des Datenmodells (in Form von Aspekten) je nach Anwendungsfall bzw. -domäne. Mit der Nutzung des Datenmodells kann zudem *ein Teil der Komplexität*, die sich aus der Variabilität der Prozesse bzw. der Konfigurierbarkeit des integrierten Modells ergibt, vor dem Anwender *verborgen* werden, was vor allem für die Darstellung von Abhängigkeiten zwischen den variantenbezogenen Entscheidungen gilt. Das Datenmodell dient zum internen Management der Varianten im Rahmen des Prozesskonfigurators. Dies wirkt sich positiv auf dessen Benutzerfreundlichkeit aus. Der Nachteil der zusätzlich notwendigen Transformation von einer bzw. in eine Anwender-orientierte Prozessmodellierungsnotation sollte daher in Kauf genommen werden. Mit einer automatisierten Lösung lässt sich zudem der Zeitaufwand der Zuordnung erheblich reduzieren und die Anzahl an Übertragungsfehlern kann im Rahmen der Qualität des Transformationsprogramms ausgeschlossen werden.

Mittels der Methodik des *gestuften Konfigurationsprozess* kann der Anwender die Auswahl seiner gewünschten Variante individuell bestimmen. Dies gilt für die Reihenfolge ebenso wie für die Anzahl der zu treffenden (Teil-) Entscheidungen an den variation points an sich, aber auch für die Entscheidungszeitpunkte, die in die Phase der Ausführung reichen können. Damit liegt ein wesentliches Unterscheidungsmerkmal zu existierenden Ansätzen vor, die sich im Wesentlichen auf die kontrollierte Durchführung dieses Vorgehens konzentrieren.

Während eine erste Evaluierung des Konfigurationskonzeptes an sich bereits mit Erfolg durchgeführt werden konnte, befindet sich die Implementierung eines entsprechenden Modellierungswerkzeugs noch in der Anfangsphase, das heißt auf Prototypebene. AOPM-Modelle als Anwender-orientierte Prozessmodelle sowie generische Datenmodelle können jedoch bereits prinzipiell in *OMME* generiert werden. Die direkte Verlinkung zwischen diesen beiden Modellen, genauer gesagt die Verlinkung der Artefakte (= Prozesselemente) der jeweiligen Modelle, ist derzeit nur über die textuelle Notation in *OMME* möglich. Die automatische Lösung ist Gegenstand aktueller bzw. zukünftiger Forschungsarbeiten. Mit Hilfe dieses ersten Prototyps konnte jedoch gezeigt werden, dass das Variantenmanagementkonzept in Bezug auf die Modellierung prinzipiell funktioniert. Auf den weiteren Forschungsbedarf wird im nachfolgenden Kapitel 7 eingegangen.

Wie die Beispiele aus den Modellierungsprojekten des Lehrstuhls und die Literaturrecherche gezeigt haben, wurde mit dem Thema der Modellierung und Konfiguration von Prozessvarianten auf eine innerhalb der Prozessdomäne anwendungsfallübergreifende Problematik eingegangen. Aufgrund der damit verbundenen Diversität der konkreten Problemstellung kann davon ausgegangen werden, dass eine Lösung, die für alle Anwendungsfälle anwendbar ist, wenn überhaupt nur sehr schwer ermittelt werden kann. Durch die nicht einzelfallbezogen, sondern anwendungsfallübergreifend durchgeführte Anforderungsanalyse kann der hier vorgestellte Prozesskonfigurator jedoch als ein in seinem Rahmen generisches Konzept gesehen werden.

Kapitel 7

Zusammenfassung

7.1 *Resumée*

Die vorliegende Arbeit motiviert sich aus der unzureichenden Durchführung der *Designphase* im Rahmen des Prozessmanagements. Die Phase des Designs dient der anforderungsgerechten Gestaltung der Prozessmodelle, was explizit von der nachfolgenden Modellierung zu trennen ist. Bereits die initiale *Anforderungsanalyse* im Rahmen des (Prozess-) Designs wird oftmals nicht mit der notwendigen Gründlichkeit und Tiefe vorgenommen; die sich anschließenden Handlungen beschränken sich meist auf die *Auswahl vorhandener (Standard-) Prozessmodellierungssprachen und –werkzeuge*; ein Customizing im Bezug auf spezifische Anforderungen findet nicht statt. Die Modellierungssprachen decken aber aufgrund ihrer festgelegten Eigenschaften nicht immer alle Anforderungen des vorliegenden Modellierungsprojektes ab. Den resultierenden Prozessmodellen fehlt es damit an Eindeutigkeit und Präzision, was sich auch negativ auf die nachfolgenden Phasen innerhalb des Prozesslebenszyklus auswirkt.

Diese allgemeine, eher abstrakte Problemstellung konnte im Bezug auf *variantenreiche Prozessmodelle* exemplarisch konkretisiert werden. Dabei handelt es sich um eine Problematik, die anwendungsübergreifend im Bereich des Prozessmanagements auftritt. Den meisten *Prozessmodellierungssprachen* fehlt es an der nötigen *Ausdrucksstärke*, sei es im Hinblick auf die Hervorhebung der variablen Stellen (variation points) in den Prozessen überhaupt oder im Bezug auf die Berücksichtigung variabler Aspekte über den Kontrollfluss hinaus. Auch können bei der Modellierung Einschränkungen bzgl. des möglichen Variantenraums oftmals nicht vorgenommen werden. Das Ableiten der Varianten aus einem integrierten, konfigurierbaren Prozessmodell stellt eine sehr *komplexe Entscheidungssituation* dar (mehrere variation points und diese bzgl. verschiedener Perspektiven, Abhängigkeiten, besondere Wünsche, fehlende Informationen), so dass es schwer ist den Überblick zu behalten. Die wird weiter dadurch erschwert, dass die Gesamtentscheidung üblicherweise zu einem fixen Zeitpunkt getroffen werden soll.

In Anbetracht der Problemstellungen war es zunächst das Ziel, eine *konzeptionelle und methodische Erweiterung des Designs* vorzunehmen. Der Fokus lag auf der Entwicklung eines methodischen Rahmenwerks, das sich nicht auf die Auswahl von Modellierungssprachen beschränkt, sondern auch die Möglichkeiten zur domänen- oder anwendungsfallspezifischen Anpassung sowie Neudefinition einer Modellierungssprache mit einbezieht. Dabei galt es explizit auf die Anforderungsanalyse einzugehen, um ziel- bzw.

umsetzungsorientiert Anforderungen an eine Prozessmodellierungssprache, Prozessmodelle oder deren Ausführung spezifizieren zu können.

Im Bezug auf die konkrete Problemstellung der variantenreichen Prozessmodelle galt es für die Modellierung eine *strukturierte, kompakte sowie holistische Darstellungsform* zu erarbeiten, die auch die Möglichkeit zur Einschränkung des Variantenraums vorsieht. Zur methodischen Unterstützung der komplexen Entscheidungssituation war es das Ziel, eine Methode zur Durchführung eines möglichst *flexiblen und variablen Konfigurationsprozesses* zu definieren, die dem Anwender dennoch ein notwendiges bzw. sinnvolles Maß an Führung bieten kann.

Für die angestrebte *konzeptionelle und methodische Erweiterung des Designs* wurde den Ausarbeitungen eine *Meta-Modell-Hierarchie* zugrunde gelegt. Diese stellt eine Methodik zur Entwicklung, Anpassung sowie Definition von verschiedenen (Meta) Modellen dar und unterscheidet zugleich zwischen Definition und Verwendung von Modellierungssprachen. Diese bisher auf die Modellierung fokussierte Meta-Modell-Hierarchie wurde um den Aspekt des Designs erweitert, wobei auch explizit auf die Anforderungsanalyse eingegangen wurde.

Das Ergebnis ist ein *Vorgehensmodell* für das Design mit enger Kopplung zu der direkt nachfolgenden Phase der Modellierung; die weiteren Phasen des Prozesslebenszyklus konnten ebenfalls mit einbezogen werden. Das Vorgehensmodell sieht explizit die Definition bzw. Anpassung von Modellierungssprachen vor. Hervorzuheben ist, dass es universell im Bezug auf Anwendungsdomäne und Modellierungssprache einsetzbar ist. Durch diese Allgemeingültigkeit ergibt sich ein breites Einsatzgebiet, welches vom Prinzip her sogar außerhalb des Bereichs der Prozessdomäne liegen kann, das heißt auf jegliche Designproblematik, so zum Beispiel innerhalb der Produktentwicklung, angewendet werden kann.

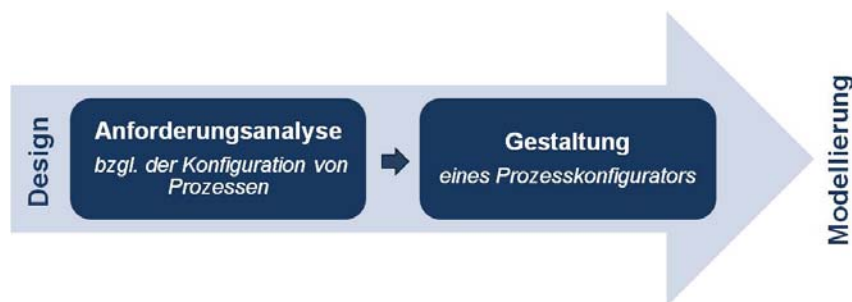


Abbildung 7-1 Konkretisierte Teilphasen des Designs

Zur Ausarbeitung des Lösungsbeitrages im Bezug auf die Modellierung und Konfiguration variantenreicher Prozessmodelle konnte das zuvor entwickelte *Vorgehensmodell für die Designphase* exemplarisch angewendet werden. Auf der Basis einer eingehenden

Anforderungsanalyse der vorliegenden Anwendungsfälle wurde das Konzept eines *Prozesskonfigurator* gestaltet (siehe Abbildung 7-1).

Im Hinblick auf die Modellierungsproblematik variantenreicher Prozesse wurde auf ein bereits existierendes Konzept zur Abbildung variantenreicher Strukturen zurückgegriffen, den sog. *mereologischen Graph*. Dieser wurde an die Domäne der Prozesse angepasst. Das Ergebnis ist ein generisches Datenmodell, das die Möglichkeit zur kompakten und strukturierten Abbildung von Prozessvarianten in einem integrierten Modell bietet. Die Notation ist unabhängig von einer konkreten Modellierungssprache und kann jederzeit um weitere als die in dieser Arbeit aufgezeigten Aspekte angepasst bzw. ergänzt werden.

Zur methodischen Unterstützung beim Ableiten der Varianten aus dem integrierten Modell wurde die bisher übliche Methode bei der Konfiguration unter Berücksichtigung der Phasen des Prozesslebenszyklus verfeinert. Die Gesamtentscheidung für eine Variante kann nun in beliebig viele Teilentscheidungen zerlegt werden, die jeweils einen oder mehrere variation points abdecken. Die Auflösungsreihenfolge kann beliebig definiert werden und der einzelne Auflösungszeitpunkt kann bis in die Phase der Ausführung hinausgezögert werden. Damit steht dem Anwender ein gestufter Konfigurationsprozess zur Verfügung, der ihn schrittweise zur gewünschten Variante leitet, ihm dabei aber maximale Handlungsfreiheit gewährt.

Mit der Problematik des Variantenmanagements wurde ein Themenbereich ausgewählt, der in zahlreichen Anwendungsdomänen (auch außerhalb der Prozessdomäne) auftritt. Aufgrund der Tatsache, dass sich dementsprechend die Detailproblematik unterscheidet, kann das Problem des Variantenmanagements nicht vollständig allgemeingültig gelöst werden. Mit dem in dieser Arbeit vorgestellten Konzept des Prozesskonfigurators soll dieser Anspruch auch nicht erhoben werden. Es ist jedoch nicht nur ein Einzelfall, wie zum Beispiel die Varianten einer Hüft-TEP-Operation, berücksichtigt worden, sondern eine begrenzte Zahl an weiteren Anwendungsdomänen. Insoweit kann von einem allgemeingültigen Ansatz des Prozesskonfigurators gesprochen werden. Die Grundlage dazu wurde durch die explizit durchgeführte Designphase gelegt, wie sie vorher in dem Vorgehensmodell herausgearbeitet worden ist. Ein entscheidender Punkt zur Erreichung des Zieles war dabei die zu Beginn explizit und umfassend durchgeführte Anforderungsanalyse über alle vorliegenden Anwendungsdomänen hinweg, auf deren Grundlage die Gestaltung des Prozesskonfigurators vorgenommen werden konnte.

Die gesamte Arbeit geht einher mit den Prinzipien der *design science research method* [HMMD04]. Dort liegt der Fokus auf der Entwicklung bzw. Konzeption zielgerichteter, realisierbarer Artefakte (Konstrukte, Modelle, Methoden oder Instanziierungen), die es in den entsprechenden Organisationen zu implementieren gilt. In der vorliegenden Arbeit wurde dementsprechend das Vorgehensmodell für die Designphase sowie das Konzept eines Prozesskonfigurators mit einem generischen, konfigurierbaren Datenmodell inklusive eines gestuften Konfigurationsprozesses ausgearbeitet.

7.2 Ausblick

Bis der erste Prototyp eines *Prozesskonfigurators* verfügbar sein wird, sind noch weitere Forschungsarbeiten bzgl. der Modellierungs- wie auch Konfigurationskomponenten vorzunehmen.

Zum einen muss das bereits in einer ersten Version als Prototyp vorhandene Modellierungswerkzeug OMME grundlegend weiter entwickelt werden. Nähere Informationen zu allgemeinen Aspekten der Weiterentwicklung sind [Volz11] zu entnehmen. Die automatische Transformation zwischen dem generischen Datenmodell und einem Anwender-orientierten Prozessmodell ist als eine essentielle Funktionalität für das Gesamtkonzept anzusehen und sollte daher möglichst zeitnah umgesetzt werden. In dem Zusammenhang wäre die Integration weiterer Modellierungssprachen neben AOPM wünschenswert, um den Anwendungsbereich zu vergrößern. BPMN und auch EPKs wären dabei an erster Stelle zu nennen, da es sich um weitverbreitete Prozessmodellierungssprachen handelt. Letztendlich ist die Funktionalität zur automatischen Überprüfung der syntaktischen und semantischen Korrektheit der Modelle sowohl für die Modellierung als auch für den Konfigurationsprozess essentiell.

Für die Konfigurationskomponente ist noch keine Implementierung des Konfigurationsmechanismus vorhanden. Konzept und Implementierung der dazugehörigen Benutzeroberfläche fehlen noch vollständig. Dabei ist die Integration zu einem Gesamtsystem mit der Modellierungsumgebung OMME oberstes Ziel und sollte daher von Beginn an umgesetzt werden.

Sind die gerade angesprochenen Punkte realisiert, ist mit dem Prozesskonfigurator eine durchgehende methodische wie werkzeugtechnische Unterstützung im Rahmen der Konfiguration variantenreicher Prozessmodelle möglich.

Im Bezug auf das Vorgehensmodell für die Designphase sollten über die Problematik variantenreicher Prozessmodelle hinaus für weitere Problemstellungen Lösungen bzw. Lösungskonzepte ausgearbeitet werden. Dabei gilt es dann der Vorgehensweise zu folgen, wie sie in dieser Arbeit präsentiert wurde; zunächst die Durchführung einer detaillierten Anforderungsanalyse, dann die darauf aufbauende Gestaltungsphase. Beispielhafte Problemstellungen sind dazu bereits in Kapitel 4 erläutert worden. Die Liste ist jedoch auf keinen Fall als vollständig zu betrachten, da sie aus der Analyse der im Rahmen dieser Arbeit zur Verfügung stehenden Projekte entstanden ist. Die Analyse weiterer Projekte, sei es in anderen oder denselben Anwendungsgebieten, wird weiteren Handlungsbedarf aufzeigen.

Literaturverzeichnis

- [AHMM02] Abecker, A., Hinkelmann, K., Maus, H. und Müller, H.J. *Geschäftsprozessorientiertes Wissensmanagement – Effektive Wissensnutzung bei der Planung um Umsetzung von Geschäftsprozessen*. Springer Verlag Berlin, 2002.
- [Agui04] Aguilar-Saven, R. Business process modeling: Review and framework. *International Journal of production economics*, 90. 129-149 2004.
- [AkKü01] Aktinson, C. und Kühne, T. *The Essence of Multilevel Metamodeling* 4th International Conference on the Unified Modeling Language, Toronto, Canada, 2001.
- [AkKü05] Aktinson, C. und Kühne, T. *Concepts for Comparing Modeling Tool Architectures* 8th International Conference MoDELS/ UML, Montego Bay, Jamaika, 2005.
- [AlGa99] Aldowaisan, T.A. und Gaafar, L.K. Business process reengineering: an approach for process mapping. *Omega, International Journal of Management Science*, 27 1999.
- [Allw05] Allweyer, T. *Geschäftsprozessmanagement – Strategie, Entwurf, Implementierung, Controlling*. W3L-Verlag, Herdecke, 2005.
- [Allw06] Allweyer, T. *Missing Links - Ein BPMS alleine macht noch kein Prozessmanagement*, BPM-Netzwerk 09. November 2006, <http://www.bpm-netzwerk.de/articles/65>, Abruf am 2010-12-01.
- [Allw10] Allweyer, T. *Unternehmen als Process Engine? Möglichkeiten und Grenzen mit BPMN*. BPMN Anwendertag 2010, Potsdam http://bpt.hpi.uni-potsdam.de/pub/BPMNAnwendertag/Program/bpmn2010_allweyer.pdf, Abruf am 2011-04-05
- [AVTa89] Alvarez-Valdés, R. und Tamarit, J.M. Heuristic algorithms for resource-constrained project scheduling: a review and an empirical analysis. In Slowinski and Weglarz eds. *Advances in project scheduling*, Elsevier Science, Amsterdam, 1989, 113-134.
- [AmSo07] Ami, T. und Sommer, R. Comparison and Evaluation of Business Process Modelling and Management Tools. *International Journal of Services and Standards*, 3 (2). 249-261 2007.
- [AnCh07] Anderer, G. und Châlons, C. *Business Process Report 2007: Geschäftsprozessmanagement in Deutschland, Österreich und der Schweiz*, 2007.
- [Andr94] Andreasen, M. Modelling - The Language of the Designer. *Journal of Engineering Design*, 5 (2). 103-105 1994.

- [ArMa97] Armistead, C.G. und Machin, S. Implications of Business Process Management for Operations Management. *International Journal of Operations & Production Management*, 17 (9). 886-898 1997.
- [ADEF04] Assmann, D., Dörr, J., Eisenbarth, M., Fichtner, H.P., Koenig, T., Nick, M., Szulmann, P. und Trifu, A. *MODALE Projektergebnisse: Stand der Technik - Notationsformen*, Fraunhofer IESE, 2004.
- [Atef97] Atefi, K. *Formal models of business process reengineering for design and design validation* M.A.Sc.Thesis, Department of Mechanical & Industrial Engineering, University of Toronto, Toronto, 1997.
- [Auto06] Autoren, A. Das V-Modell®XT. Bundesrepublik Deutschland, 2006.
- [Ball70] Ballauf, T. *Skeptische Didaktik*. Quelle & Meyer, Heidelberg, 1970.
- [BICS07] Bandara, W., Indulska, M., Chong, S. und Sadiq, S. *Major Issues in Process Management: An Expert Perspective*. The 15th European Conference on Information Systems (ECIS'07), St.Gallen, Switzerland, 2007, 1240-1251.
- [Barr05] Barros, O. A novel approach to joint business and information system design *Journal of Computer and Information System*, 45 (3). 96-106 2005.
- [Baum96] Baumgarten, B. *Petri-Netze - Grundlagen und Anwendung*. Spektrum-Akademischer Verlag, Heidelberg, 1996.
- [BaBG05] Bayer, J., Buhl, W., Giese, C., Lehner, T., Ocampo, A., Puhlmann, F., Richter, E., Schnieders, A., Weiland, A. und Weske, M. *PESOA - Process Family Engineering - Modeling variant-rich processes*. Engineering, F.I.E.S. ed., Hasso-Plattner-Institut, Potsdam, Germany, 2005.
- [BDDK04] Becker, J., Delfmann, A., Dreiling, A., Knackstedt, R. und Kuropka, D. *Configurative Business Process Modeling - Outlining an Approach to increased Business Process Model Usability* 14th Information Resources Management Association Int'l. Conference (IRMA '04), New Orleans, USA, 2004.
- [BeDK07] Becker, J., Delfmann, P. und Knackstedt, R. *Adaptive Reference Modeling: Integrating Configurative and Generic Adaption Techniques for Information Models* in Becker, J. and Delfmann, P. eds. *Reference Modeling* Springer Verlag, Berlin, 2007, 27-58.
- [BDKK02] Becker, J., Delfmann, A., Knackstedt, R. und Kuropka, D. *Konfigurative Referenzmodellierung*. In *Wissensmanagement mit Referenzmodellen*, Physica Verlag, Heidelberg, 2002.
- [BKKD01] Becker, J., Knackstedt, R., Kuropka, D. und Delfmann, P. *Subjektivitätsmanagement für die Referenzmodellierung: Vorgehensmodell und Werkzeugkonzept* KnowTech, Dresden, 2001.

- [BeKR05] Becker, J., Kugeler, M. und Rosemann, M. *Processmanagement*, Springer Verlag, Berlin, 2005.
- [BeRS95] Becker, J., Rosemann, M. und Schütte, R. Grundsätze ordnungsmäßiger Modellierung. *Wirtschaftsinformatik*, 37 (5). 435-445 1995.
- [BeRU00] Becker, J., Rosemann, M. und Uthmann, C. Guidelines of Business Process Modeling. in van der Aalst, W.M.P., Desel, J. und Oberweis, A. eds. *Business Process Management. Models, Techniques, and Empirical Insights*, Springer Verlag, Berlin, 2000
- [BiIn09] Bibliographisches Institut. *Duden*. Bibliographisches Institut AG, Mannheim 2009.
- [Birk06] Birker, K. *Projektmanagement - Lehr- und Arbeitsbuch für die Aus- und Weiterbildung*. Cornelsen Verlag, Berlin, 2006.
- [BAKF04] Blecker, T., Abdelkafi, N., Kreuter, G. und Friedrich, G. *Product Configuration Systems: State-of-the-Art, Conceptualization and Extensions* Génie logiciel & Intelligence artificielle. 8th Magrebian Conference on Software Engineering and Artificial Intelligence (MCSEAI2004), Sousse, Tunesia, 2004.
- [BIWB03] Blin, M.-J., Wainer, J. und Bauzer-Medeiros, C. A Reuse-Oriented Workflow Definition Language. *International Journal of Cooperative Information Systems*, 12 (1). 1-36 2003.
- [Bode06] Bodendorf, F. *Daten- und Wissensmanagement*. Springer Verlag, Berlin, 2006.
- [BoRB05] Bobrik, R., Reichert, M. und Bauer, T. *Requirements for visualization of system-spanning business process* DEXA, 2005, 948-954.
- [Böhm00] Böhm, M. *Entwicklung von Workflow-Typen - Ein Leitfaden der methodischen Anwendungsentwicklung am Beispiel ausgewählter Workflow-Aspekte*. Springer Verlag, Berlin, 2000.
- [Bont04] Bontrup, H.-J. *Volkswirtschaftslehre - Grundlagen der Mikro- und Makroökonomie*. Oldenbourg Wissenschaftsverlag GmbH, München, 2004.
- [Broc93] Brockhaus, F.A. *Brockhaus-Enzyklopädie*, Brockhaus, Wiesbaden, 1993.
- [BrSt04] Broy, M. und Steinbrüggen, R. *Modellbildung in der Informatik*. Springer Verlag, Berlin, 2004.
- [Burg02] Burghardt, M. *Projektmanagement - Leitfaden für die Planung , Überwachung und Steuerung von Entwicklungsprojekten* Publicis Corporate Publishing, Erlangen, 2002.
- [BuIn06] Bundesministerium des Innern. *V-Modell XT, Version 1.3 (english)*, 2006. <http://v-modell.iabg.de/dmdocuments/V-Modell-XT-Gesamt-Englisch-V1.3.pdf>, Abruf am 2010-09-09

- [BuBF11] Bundesministerium für Bildung und Forschung. Der Bologna-Prozess. <http://www.bmbf.de/de/3336.php>, Abruf am 2011-03-28
- [Buss98] Bussler, C. Organisationsverwaltung in Workflow-Management-Systemen. Deutscher Universitäts-Verlag Wiesbaden, 1998.
- [BuAn89] Buur, J. und Andreassen, M. Design models in mechatronic product development. *Design Studies* (3). 105-162 1989.
- [Butl96] Butler, P. *A strategic framework for health promotion in Darebin*. Center for development and innovation Melbourne, Australia 1996.
- [CKKP06] Caupin, G., Knöpfel, H., Koch, G., Pannenbäcker, K., Pérez-Polo, F. und Seaburg, C. *ICB – IPMA Competence Baseline Version 3.0*. IPMA International Project Management Association, Nijkerk, Netherland, 2006.
- [Chen76] Chen, P.P.-S. The Entity-Relationship Model - Towards a Unified View of Data *ACM Transactions on Database Systems*, 1 (1). 9-36 1976.
- [CISW08] Clark, T., Sammut, P. und Willans, J. *Applied Meta Modeling - A Foundation for language driven development* Ceteva, 2008.
- [Clem06] Clements, P.C. *Managing Variability for Software Product Lines: Working with Variability Mechanisms* 10th Int'l. Conference on Software Product Lines (SPLC'06), Baltimore, Maryland, 2006.
- [CINo02] Clements, P. and Northrop, L. *Software Product Lines: Practices and Patterns*. Addison-Wesley, Upper Saddle River, NJ, 2002.
- [CuKO92] Curtis, B., Kellner, M.I. und Over, J. Process Modeling. *Communications of the ACM*, 35 (9) 1992.
- [CuKe97] Curran, T. und Keller, G. *SAP R/3 Business Blueprint: Understanding the Business Process Reference Model* Upper Saddle River, 1997.
- [CzHE04] Czarnecki, K., Helsen, S. und Eisenecker, U. *Staged Configuration Using Feature Models* 3rd Int'l. Conference on Software Product Lines (SPLC'04), Boston, MA, USA, 2004.
- [CzHE05] Czarnecki, K., Helsen, S. und Eisenecker, U. Staged Configuration Through Specialization and Multi-Level Configuration of Feature Models. *Software Process: Improvement and Practice*, 10 (1) 2005.
- [DaRK00] Dadam, P., Reichert, M. und Kuhn, K. *Clinical Workflows - The Killer Application for Process-Oriented Information Systems?* 4th Int'l Conference on Business Information Systems (BIS '00) Poznan, Poland, 2000.

- [DaKo05] Danesh, A. und Kock, N. An experiment study of process representation approaches and their impact on perceived modeling quality and redesign success *Business Process Management Journal*, 11 (5). 724-735 2005.
- [DaSh96] Darke, P. und Shanks, G. Stakeholder viewpoints in requirements definition: A framework for understanding viewpoint development approaches *Requirements Engineering*, 1 (2). 88-105 1996.
- [DaSh90] Davenport, T.H. und Short, J.E. The New Industrial Engineering: Information Technology and Business Process Redesign. *Sloan Management Review* 31 (4). 11-27 1990.
- [Dave93] Davenport, T.H. Process Innovation– Reengineering Work Through Information Technology. *Harvard Business School Press* Boston, Massachusetts, 1993.
- [Dave05] Davenport, T.H. The Coming Commoditization of Processes. *Harvard Business Review* 83 (6). 100-108 2005.
- [DGRI06] Davies, I., Green, P., Rosemann, M., Indulska, M. und Gallo, S. How do practitioners use conceptual modeling in practice? *Data & Knowledge Engineering*, 58 (3). 358-380 2006.
- [DevA04] Dehnert, J. und van der Aalst, W.M.P. Bridging The Gap Between Business Models And Workflow Specifications. *International Journal of Cooperative Information Systems* 13. 289-332 2004.
- [Denn97] Denning, P.J. A New Social Contract for Research. *Communications of the ACM*, 40 (2). 132-134 1997.
- [DeIN09] Deutsches Institut für Normung e.V. (DIN) *Projektmanagement, Projektmanagementsysteme (DIN 69901)*. 2009
- [DeIN10] Deutsches Institut für Normung e.V. (DIN), <http://www.din.de>, 2010.
- [DeIn02a] Deutsches Institut für Normung e.V. (DIN) *Begriffe für Stücklisten und das Stücklistenwesen (DIN 199-2)*. 2002.
- [DeIn09] Deutsches Institut für Normung e.V. (DIN) DIN 69901 Projektmanagement – Projektmanagementsysteme. 2009
- [DeTV95] Deutscher Taschenbuch-Verlag. *dtv-Lexikon*. Deutscher Taschenbuch-Verlag, München, 1995.
- [Dict10] dict.cc, <http://www.dict.cc>, Abruf am 2010-09-24
- [DSMF10] DSM Forum DSM Domain-Specific Modeling. <http://www.dsmforum.org/>, Abruf am 2010-11-22, 2010

- [DuAH05] Dumas, M., van der Aalst, W.M.P. und ter Hofstede, A. *Process Aware Information Systems: Bridging People and Software Through Process Technology* John Wiley & Sons, Hoboken, NJ, USA, 2005.
- [Dym94] Dym, C.L. *Engineering Design* Cambridge University Press New York, 1994.
- [DyLi00] Dym, C.L. und Little, P. *Engineering Design: A Project-Based Introduction*. John Wiley & Sons, New York, 2000.
- [EcFI11] Eclipse Foundation Inc. *Eclipse*. <http://www.eclipse.org/>, 2011. Abruf am 2011-04-17.
- [Ehrl07] Ehrlenspiel, K. *Integrierte Produktentwicklung - Denkabläufe, Methodeneinsatz, Zusammenarbeit*. Carl Hanser Verlag, München, 2007.
- [EHLB95] Elzinga, D.J., Horak, D., Lee, C.-Y. und Bruner, C. Business Process Management: Survey and Methodology. *IEEE Transaction on Engineering Management*, 42 (2). 119-128 1995.
- [Erl05] Erl, T. *Service-oriented Architecture: Concepts, Technology, and Design* Prentice Hall, Upper Saddle River, New Jersey 2005.
- [FMSJ09] Faerber, M., Meerkamm, S., Schneider, T. und Jablonski, S. Qualitative Prozessanalyse klinischer Prozesse - Vorgehen und Erfahrung. *arthritis + rheuma*, 29 (6). 353-359 2009.
- [Fähn09] Fähndrich, S. *Handbuch Qualitätssicherung und Qualitätsentwicklung zur Einrichtung und Weiterentwicklung von Studiengängen an der Johannes Gutenberg-Universität Mainz*. Johannes Gutenberg-Universität Mainz, Mainz, 2009.
- [FeBD90] Fetter, R.B., Brand, D.A. und Garnache, D. *DRGs: Their Design and Development*. Health Administration Press, 1990.
- [FeLo03] Fettke, P. und Loose, P. Classification of reference models: a methodology and its application *Information Systems and e-Business Management*, 1. 35-53 2003.
- [FeLZ05] Fettke, P., Loos, P. und Zwicker, J. *Business Process Reference Models: Survey and Classification* Workshop on Business Process Reference Models (BPRM'05), Nancy, France, 2005.
- [FiHR08] Fieber, F., Huhn, M. und Rumpe, B. Modellqualität als Indikator für Softwarequalität: eine Taxonomie. *Informatik Spektrum*, 31 (5). 17 2008.
- [Fisc01] Fischer, W. *Grundzüge von DRG-Systemen - Krankenhausreport 2000*. Schattauer Verlagsgesellschaft, 2001.
- [Fisc06] Fischermanns, G. *Praxishandbuch Prozessmanagement*. Verlag Dr. Götz Schmidt, Gießen, 2006.

- [Ford22] Ford, H. *My life and work*. Garden City, 1922 (Nachdruck: New York, 1973), New York, 1922/1973.
- [FrPr97] Frank, U. und Prasse, M. Zur Standardisierung objektorientierter Modellierungssprachen: Eine kritische Betrachtung des State of the Art am Beispiel der Unified Modeling Language. *Rundbrief des GI-Fachausschusses*, 5.2 (1). 1-5 1997.
- [Fran98] Frank, U. *Evaluating Modelling Languages: Relevant Issues, Epistemological Challenges and a Preliminary Research Framework* Arbeitsbericht des Instituts für Wirtschaftsinformatik, Koblenz, 1998.
- [FrvL03] Frank, U. und van Laak, B.D. Anforderungen an Sprachen zur Modellierung von Geschäftsprozessen. Frank, U., Hampe, F. ed. *Arbeitsbericht des Instituts für Wirtschaftsinformatik, Institut für Wirtschaftsinformatik*, Universität Koblenz-Landau, Koblenz-Landau, 2003.
- [FrRH10] Freund, J., Rücker, B. und Henninger, T. *Praxishandbuch BPMN*, Carl Hanser Verlag, München, 2010.
- [Funk03] Funke, J. *Problemlösendes Denken*. Kohlhammer Verlag, Stuttgart, 2003.
- [Gab110] Gabler. *Gabler Wirtschaftslexikon*. <http://wirtschaftslexikon.gabler.de/>, Abruf am 2010-09-09
- [Gada05] Gadatsch, A. *Grundkurs Geschäftsprozessmanagement. Methoden und Werkzeuge für die IT-Praxis: Eine Einführung für Studenten und Praktiker*. Vieweg Verlag, Wiesbaden, 2005.
- [Gair81] Gairola, A. *Montagegerechtes Konstruieren - Ein Beitrag zur Konstruktionsmethodik* Fachbereich Nachrichtentechnik, Technische Hochschule Darmstadt, Darmstadt, 1981.
- [Gait83] Gaitanides, M. *Prozessorganisation. Entwicklung, Ansätze und Programm prozessorientierter Organisationsgestaltung*. Vahlen Verlag, München 1983.
- [GaSc95] Galler, J. und Scheer, A.-W. Workflow-Projekte: Vom Geschäftsprozessmodell zur unternehmensspezifischen Workflow-Anwendung. *Information Management* (10). 20-27 1995.
- [GaGr05] Gartner Group. *Delivering IT's Contribution: The 2005 CIO Agenda. EXP Premier Report Nr. January 2005*, Gartner Inc., Stamford, Connecticut, 2005.
- [GaGr06] Gartner Group. *Growing IT's Contribution: The 2006 CIO Agenda. EXP Premier Report Nr. January 2006*, Gartner Inc., Stamford, Connecticut, 2006.
- [GaGr07] Gartner Group. *Creating Enterprise Leverage: The 2007 CIO Agenda. EXP Premier Report Nr. January 2007*, Gartner Inc., Stamford, Connecticut, 2007.

- [GaGr09] Gartner Group. *Meeting the Challenge: The 2009 CIO Agenda EXP Premier Report January 2009*, Gartner Inc., Stamford, Connecticut, 2009.
- [GeWa97] Geib, T. und Wagner, K. *Neue Wege der Geschäftsprozessgestaltung*. Information Management & Consulting, 12 (Sonderausgabe). 79-82 1997.
- [GeKP98] Geisler, R., Klar, M. und Pons, C. *Dimensions and Dichotomy in Metamodeling*. Technical Report 98-5 Technical University Berlin, 1998.
- [GeHS95] Georgakopoulos, D., Hornick, M. und Shet, A. An Overview of Workflow Management: From Processes Modeling to Workflow Automation Infrastructure. *Distributed and Parallel Database*, 3 (2). 35 1995.
- [Giag01] Giaglis, G.M. A taxonomy of business process modeling and information system modeling techniques. *International Journal of Flexible Manufacturing Systems*, 13 (2). 209-228 2001.
- [GiGW00] Giese, H., Graf, J. und Wirtz, G. *Präzisierung des UML-Metamodells durch ein semantisches Objektmodell* GI Softwaretechnik-Trends 20(2), Beitrag für den 7-ten GROOM-Workshop "UML – Erweiterungen (Profile) und Konzepte der Metamodellierung", Koblenz-Landau, Germany, 2000.
- [Glas99] Glass, R. On Design. *IEEE Software*, 16 (2). 103-104 1999.
- [Golo97] Golovin, J. *Achieving Stretch Goals: Best Practice in Manufacturing for the New Millennium*. Prentice-Hall, New York, 1997.
- [GoLa08] Gottschalk, F. und La Rosa, M. Process Configuration in YAWL *QUT ePrints 15718*, Queensland University of Technology, 2008.
- [GoAJ08] Gottschalk, F., van der Aalst, W.M.P., Jansen-Vuller, M.H. und LaRosa, M. Configurable Workflow Models *International Journal of Cooperative Information Systems*, 17 (2). 223-255 2008.
- [Götz10] Götz, M. *Entwicklung Domänenspezifischer Software - Dargestellt am Beispiel des Prozessmanagements* Dissertation, Fakultät für Mathematik, Physik und Informatik, Universität Bayreuth, Bayreuth, 2010.
- [GrvH00] Gräfin vom Hagen, Ramona. *Die Universität – ein Unternehmen?*. http://www.uni-heidelberg.de/presse/ruca/ruca1_2000/vom_hagen.html, Abruf am 2011-03-28
- [GrRe93] Gray, J. und Reuter, A. *Transaction Processing – Concepts and Techniques*. Morgan Kaufmann Publishers, Inc., San Fransisco, 1993.
- [GrKo05] Großmann, M. und Koschek, H. *Unternehmensportale - Grundlagen, Architekturen, Technologien*. Springer Verlag, Berlin, 2005.
- [Hein99] Heina, J. *Variantenmanagement - Kosten-Nutzenbewertung zur Optimalen Variantenvielfalt*. Deutscher Universitäts-Verlag, Wiesbaden 1999.

- [HaBR08a] Hallerbach, A., Bauer, T. und Reichert, M. *Anforderungen an die Modellierung und Ausführung von Prozessvarianten*. Datenbank Spektrum 2008.
- [HaBR08b] Hallerbach, A., Bauer, T. und Reichert, M. *Issues in Modeling Process Variants with Provop* 4th International Workshop on Business Process Design (BPD'08), Milan, Italy, 2008.
- [HaBR08c] Hallerbach, A., Bauer, T. und Reichert, M. *Managing Process Variants in the Process Life Cycle* 10th Int'l Conf. on Enterprise Information Systems (ICEIS'08), Barcelona Spain, 2008.
- [HaBR08d] Hallerbach, A., Bauer, T. und Reichert, M. *Modellierung und Darstellung von Prozessvarianten in PROVOP* Modellierung'08 Conference, Berlin, Germany, 2008.
- [HaBR10] Hallerbach, A., Bauer, T. und Reichert, M. Capturing Variability in Business Process Models: The Provop Approach. *Journal of Software Maintenance and Evolution: Research and Practice*, 22 (6-7). 519-546 2010.
- [Hamm90] Hammer, M. Reengineering Work: Don't Automate, Obliterate. *Harvard Business Review*, 68 (4). 104-112 1990.
- [Hamm97] Hammer, M. *Beyond Reengineering: How the Process-Centered Organization is Changing Our Work and Our Life*. HarperBusiness, New York, 1997.
- [Hamm01] Hammer, M. The Superefficient Company. *Harvard Business Review*, 79 (September-October). 82-91 2001.
- [HaCh93] Hammer, M. und Champy, J. *Reengineering the Cooperation: A Manifesto for Business Revolution*. Nicholas Brealy Publishing, London, 1993.
- [HaCh03] Hammer, M. und Champy, J. *Business Reengineering – Die Radikalkur für das Unternehmen*. Campus, Frankfurt/ New York, 2003.
- [HaRu04] Harel, D. und Rumpe, B. Meaningful modeling: What's semantics of "semantics"? *Computer*, 37 (10). 64-72 2004.
- [Hars94] Hars, A. *Referenzdatenmodelle: Grundlagen effizienter Datenmodellierung*. Gabler Verlag, Wiesbaden, 1994.
- [Hell02] Hellmann, W. *Klinische Pfade: Konzepte, Umsetzung, Erfahrung*. ecomed, Landsberg/ Lech, 2002.
- [HeGo06] Henderson-Seller, B. und Gonzalez-Perez, C. A power-type-based metamodeling framework. *Software and System Modeling*, 5 (1). 72-90 2006.
- [Hern05] Hernandez, R. *The SAP R/3 Handbook*. McGraw-Hill & Osborne Media, 2005.
- [HeMa08] Hesse, W. und Mayr, H. Modellierung in der Softwaretechnik - eine Bestandsaufnahme. *Informatik Spektrum*, 31 (5). 377-393 2008.

- [HMPR04] Hevner, A.R., March, S.T., Prak, J. und Ram, S. Design Science in Information Systems Research. *MIS Quarterly* 28. 75-105 2004.
- [HoVe01] Hofacker, I. und Vetschera, R. Algorithmical Approaches to Business Process Design *Computer & Operational Research* 28 (13). 1253-1275 2001.
- [Hofe99] Hofer-Alfeis, J. *Geschäftsprozessmanagement - innovative Ansätze für wandlungsfähige Unternehmen* Tectum Verlag, 1999.
- [HSMR02] Holler, T., Schmid, K., Müller, H.P., Reemts, C., Bissat, K. und Rieben, E. *Praktische Pfadarbeit - Konstruktion, Implementierung und Controlling von Patientenpfaden*. in Hellmann, W. ed. *Klinische Pfade*, Ecomed, Landsberg, 2002, 38-129.
- [Horn03] Horn, S. *Die schemabasierte Modellierung und Steuerung von Projektvorgängen*. Dissertation. Universität Erlangen-Nürnberg, Erlangen, 2003.
- [HuSW04] Huan, S.H., Sheoran, S.K. und Wang, G. A Review and Analysis of Supply Chain Operations Reference (SCOR) Model. *Supply Chain Management - An International Journal*, 9 (1). 23-29 2004.
- [Hümm04] Hümmel, W. *Vertragsverhandlungen um konfigurierbare Produkte im elektronischen Handel* Technische Fakultät, Dissertation. Erlangen Nürnberg, Erlangen 2004.
- [HMMD04] Hümmel, W., Meiler, C., Müller, S. und Dietrich, A. *Data Model and Personalized Configuration Systems for Mass Customization - A Two Step Approach for Integrating Technical and Organizational Issues*. Int'l Conference on Economic, Technical and Organizational aspects of Product Configuration Systems, Copenhagen, Denmark, 2004.
- [Hung06] Hung, R. Business process management as a competitive advantage: a review and empirical study. *Total Quality Management & Business Excellence*, 17 (1). 21-40 2006.
- [IBM11] IBM. IBM Rational Software Architect. <http://www.ibm.com/developerworks/rational/products/rsa/>, Abruf am 2011-03-28.
- [IcSZ93] Icmeli, O., Selcuk, S.E. und Zappe, C.J. Project scheduling problems: a survey. *International Journal of Operations and Production Management*, 13. 80-91 1993.
- [IDSS10a] IDS Scheer. ARIS ITIL. [http://www.ids-scheer.com/de/ARIS/ARIS_Referenzmodelle/ARIS_ITIL/105129.html?mod_dcert\[aching\]=953534932](http://www.ids-scheer.com/de/ARIS/ARIS_Referenzmodelle/ARIS_ITIL/105129.html?mod_dcert[aching]=953534932), Abruf am 2011-05-25, 2010
- [IDSS10b] IDS Scheer: ARIS Plattform. http://www.ids-scheer.com/de/ARIS/ARIS_Plattform/105123.html, Abruf am 2011-05-25, 2010.

- [IDSS10c] IDS Scheer. *ARIS SCOR*. http://www.ids-scheer.com/de/ARIS/ARIS_Referenzmodelle/SCOR/103795.html, Abruf am 2011-05-25, 2010.
- [IJMZ10] Igler, M., Jablonski, S., Moura, P. und Zeising, M. *ESProNa: Constraint-Based Declarative Business Process Modeling* 3rd Int'l Workshop on Dynamic and Declarative Business Processes (DDBP 2010), Vitória, ES, Brazil 2010.
- [IGRR09] Indulska, M., Green, P., Recker, J. und Rosemann, M. *Business process modeling: perceived benefits* 28th International Conference on Conceptual Modeling, Gramado, Brazil, 2009.
- [InRe08] Indulska, M. and Recker, J. *Design Science in IS Research* 4th Biennial ANU Workshop on Information System Foundations, Canberra, Australia, 2008.
- [IRRG09] Indulska, M., Recker, J., Rosemann, M. und Green, P. *Business Process Modeling: current issues and future challenges* 21st International Conference on Advanced Information Systems, Amsterdam, The Netherlands, 2009.
- [IBM11] International Business Machines (IBM). *Business Process Execution Language for Web Services, Version 1.1*. <http://www.ibm.com/developerworks/library/specification/ws-bpel/>, Abruf am 2011-05-25, 2011
- [Jabl94] Jablonski, S. *MOBILE: A Modular Workflow Model and Architecture* 4th Int'l. Working Conference on Dynamic Modeling and Information Systems Noordwijkerhout, NL, 1994.
- [Jabl95] Jablonski, S. *Workflow Management Systeme: Modellierung und Architektur*. International Thomson Publishing, Bonn 1995.
- [Jabl09] Jablonski, S. *Process Modeling for Holistic Process Management*. in Cardoso, J. und van der Aalst, W.M.P. eds. *Handbook of Research on Business Process Modeling*, Information Science Reference, 2009.
- [Jabl10] Jablonski, S. Do We Really Know How To Support Processes? Considerations and Reconstruction. *Graph Transformations and Model-Driven Engineering* 2010. 393-410 2010.
- [JaBS99] Jablonski, S., Böhm, M. und Schulze, W. *Workflow Management. Entwicklung von Anwendungen und Systemen - Facetten einer neuen Technologie*, dpunkt Verlag, Heidelberg, 1999.
- [JaBu96] Jablonski, S. und Bussler, C. *Workflow Management – Modeling Concepts, Architecture and Implementation*. International Thomson Computer Press, London, 1996.

- [Jabl01] Jablonski, S. Von der Anwendungsanalyse zu ersten Systemkonzepten für Workflow-Management-Lösungen *Bericht des Transferbereichs 21: Operationales Prozess- und Datenmanagement für Großprojekte im Anlagenbereich, Arbeitsberichte des Instituts für Informatik* Erlangen, 2001.
- [JaDV08] Jablonski, S., Dornstauder, S. and Volz, B. *A Meta Modeling Framework for Domain Specific Process Management*. 1st IEEE Int'l Workshop on Semantics for Business Process Management (SemBPM'08), Turku, Finland, 2008.
- [JaTa09] Jablonski, S. und Talib, R. *Agent Assignment for Process Management: Pattern Based Agent Performance Evaluation* 4th Int'l. Workshop on Agents and Data Mining Interaction (ADMI 2009), Budapest, Hungary, 2009.
- [JaVD09] Jablonski, S., Volz, B. and Dornstauder, S. *Evolution of Business Process Models and Languages* 2nd International Conference on Business Process and Services Computing (BPSC'09), Leipzig, 2009.
- [jBen10] jBoss Enterprise. *jBPM*. <http://www.jboss.org/jbpm/>, Abruf am 2010-11-22. 2010.
- [JMPW93] Johansson, H.J., McHugh, P., Pendlebury, A.J. und Wheeler, W.A. *Business Process Reengineering: BreakPoint Strategies for Market Dominance*. John Wiley & Sons, Chichester, 1993.
- [KaKü02] Karagiannis, D. und Kühn, H. *Metamodelling Platforms* 3rd Int'l. Conference EC-Web. Aix-en-Provence, France, 2002.
- [KeNS92] Keller, G., Nüttgens, M. und Scheer, A.-W. Semantische Prozessmodellierung auf der Grundlage Ereignisgesteuerter Prozessketten (EPK) *Veröffentlichungen des Instituts für Wirtschaftsinformatik - Heft 98*, Saarbrücken 1992.
- [KiKr08] Kipp, T. und Krause, D. *Design for variety - efficient support for design engineers* 10th International Design Conference - Design 2008, Dubrovnik, 2008, 425-432.
- [Köhl06] Köhler, P.T. *Prince2 - Das Projektmanagement-Framework*. Springer Verlag, Berlin 2006.
- [KrRV07] Krahn, H., Rumpe, B. und Völkel, S. *Integrated Definition of Abstract and Concrete Syntax for Textual Languages in Model Driven Engineering Languages and Systems*, Springer Verlag, Berlin, 2007.
- [KELR10] Krehmer, H., Eckstein, R., Lauer, W., Roelofsen, J., Stöber, C., Troll, A., Zapf, J., Weber, N., Meerkamm, H., Henrich, A., Lindemann, U., Rieg, F. und Wartzack, S. Das Forflow-Prozessmodell zur Unterstützung der multidisziplinären Produktentwicklung. *Konstruktion 10/2010*. 59-68 2010.
- [Kuen00] Kueng, P. The effects of Workflow Systems in Organization: A Qualitative Study. In van der Aalst, W., Desel, J. und Oberweis, A. eds. *Business Process Management*, Springer Verlag, Berlin, 2000, 301-315.

- [KBGS11] Kurbel, K., Becker, J., Gronau, N., Sinz, E. und Suhl, L. *Enzyklopädie der Wirtschaftsinformatik - Online Lexikon*, <http://www.enzyklopaedie-der-wirtschaftsinformatik.de>, 2011. Abruf am 2011-04-17.
- [Lamn95] Lamnek, S. *Qualitative Sozialforschung*. Springer Verlag, Weinheim, 1995.
- [LaRo08a] La Rosa, M. *Managing variability in process-aware information systems*, Queensland University of Technology, Brisbane, 2008.
- [LaRo10] La Rosa, M. *Process Configuration*, <http://www.processconfiguration.com/> Abruf am 2011-05-26, 2010
- [LDHM11] La Rosa, M., Dumas, M., ter Hofstede, A. und Mendling, J. Configurable Multi-Perspective Business Process Models. *Information Systems*, 36 (2). 313-340 2011.
- [LaGD07] La Rosa, M., Gottschalk, F., Dumas, M. und van der Aalst, W.M.P. *Linking Domain Models and Process Models for Reference Model Configuration* 10th Int'l Workshop on Reference Modeling, Brisbane, Australia, 2007.
- [LLSD07] La Rosa, M., Lux, M., Seidel, S., Dumas, M. und Hofstede ter, A. *Questionnaire driven Configuration of References Models* 19th International Conference on Advanced Information Systems Engineering (CAiSE'07), Trondheim, Norway, 2007.
- [LaVi01] Lauesen, S. und Vinter, O. Preventing Requirement Defects: An Experiment in Process Improvement *Requirements Engineering*, 6. 37-50 2001.
- [Laut11] Lautenbacher, T. *Die Entwicklung von Geschäftsideen - Ein Leitfaden zur systematischen Erzeugung, Bewertung und Auswahl von Ideen für neue Geschäftsfelder im Rahmen des Internal Corporate Venturing*. VDM, Saarbrücken, 2011.
- [LeDa98] Lee, R.G. und Dale, B.G. Business Process Management: A Review and Evaluation. *Business Process Management Journal*, 4 (3). 214-225 1998.
- [LeMR94] Leu, Y.-Y., Matheson L.A. und Rees, L.P. Assembly line balancing using genetic algorithms with heuristic - generated initial populations and multiple evaluation criteria. *Decision Sciences*, 25. 581-606 1994.
- [LBHH05] Lenz, R., Buessecker, F., Herlofsen, H., Hinrichs, F., Zeiler, T. und Kuhn, A.H. *Demand-Driven Evolution of IT-Systems in Healthcare - A case Study for Improving Interdisciplinary Processes Methods* Inf Med, Schattauer Verlag, 44. 4-20 2005.
- [Levi02] Levitin, A. *Introduction to the Design & Analysis of Algorithms*. Addison Wesley, USA, 2002.
- [LiPr96] Liles, DAS HEIßT und Presley, A.R. *Enterprise modeling within an enterprise engineering framework* Proceedings of the 28th conference on Winter simulation, Coronado, California, United States, 1996, 993-999.

- [LiRe05] Liman, M.S. und Reijers, H.A. Best Practice in Business Process Redesign: Validation of a Redesign Framework. *Computers in Industry*, 56 (5). 457-471 2005.
- [LiSS94] Lindland, O.I., Sindre, G. und Sølvsberg, A. Understanding Quality in Conceptual Modeling. *IEEE Software*, 11 (2). 42-49 1994.
- [LPZE07] LPZ E-Business. Domänenspezifische Sprachen. *LPZ – IT- Radar*, 10 (2) 2007.
- [LuSa06a] Lu, R. und Sadiq, S. *Managing Process Variants as an Information Resource* 4th Int'l. Conference on Business Process Management (BPM '06), Vienna, Austria, 2006.
- [LuSa06b] Lu, R. und Sadiq, S. *On Managing Process Variants as an Information Resource*, Technical Report No. 464, School of Information Technology and Electrical Engineering, The University of Queensland Brisbane, 2006.
- [MCLP99] Malone, T.W., Crowstone, K., Lee, J., Pentland, B., Dellarocas, C., Wyner, G., Quimby, J., Osborn, C., Bernstein, A., Herman, G., Klein, M. und O'Donnell, E. Tools for Inventing Organizations: Towards a Handbook for Organizational Processes. *Management Science*, 45 (3) 1999.
- [MaSm95] March, S.T. und Smith, G.F. Design and natural science research on information technology *Decision Support Systems*, 15 (4). 251-266 1995.
- [MaMG02] Markus, M.L., Majchrzak, A. und Gasser, L. A Design Theory for Systems that Support Emergent Knowledge Processes. *MIS Quarterly*, 26 (3). 179-212 2002.
- [Maur96] Maurer, G. *Von der Prozessorientierung zum Workflow Management, Teil 2: Prozessmanagement, Workflow Management, Workflow-Management-Systeme* Arbeitspapiere WI, Lehrstuhl für Allg. BWL und Wirtschaftsinformatik, Johannes Gutenberg Universität, Mainz, 1996.
- [MaSc97] Maurer, G. and Schwickert, A.C. *Kritische Anmerkung zur Prozessorientierung*. Arbeitspapiere WI, 9 1997.
- [Maye06] Mayer, H. *Interview und schriftliche Befragung*. Oldenbourg Wissenschaftsverlag, Oldenbourg, 2006.
- [Meer05] Meerkamm, H. Integrierte Produktentwicklung *Bildband zur Vorlesung*, Universität Erlangen-Nürnberg, Erlangen, 2005.
- [Meer07] Meerkamm, H. Prozesse: eine aktuelle Herausforderung in der Produktentwicklung. *Konstruktion*, 9. 1 2007.
- [Meer07b] Meerkamm, S. *Konzeption eines Prozessnavigators zum integrierten Prozess- und Projektmanagement in der Produktentwicklung*. Diplomarbeit, Technische Fakultät der Universität Erlangen-Nürnberg Erlangen, 2007.

- [MePa06] Meerkamm, H. und Paetzold, K. *FORFLOW - Bayerischer Forschungsverbund für Prozess- und Workflowunterstützung zur Planung und Steuerung der Abläufe in der Produktentwicklung (Langantrag)*. Print GmbH, München, 2006.
- [MePa08] Meerkamm, H. und Paetzold, K. *FORFLOW - Bayerischer Forschungsverbund für Prozess- und Workflowunterstützung zur Planung und Steuerung der Abläufe in der Produktentwicklung (2. Ergebnisbericht)* Print GmbH, München, 2008.
- [Meer09] Meerkamm, S. *The Concept of Process Management in Theory and Practice - A Qualitative Analysis* 1st. Int'l. Workshop on Empirical Research in Process Management (ER-BPM '09) Ulm, Germany, 2009.
- [Meil05] Meiler, C. *Modellierung, Planung und Ausführung Klinischer Pfade* Dissertation, Technische Fakultät der Universität Erlangen-Nürnberg Erlangen, 2005.
- [Mend08] Mendling, J. *Metrics for Process Models: Empirical Foundations of Verification, Error Prediction, and Guidelines for Correctness*. Lecture Notes in Business Information Processing, Volume 6, Springer Verlag, Berlin, 2008.
- [MeRA10] Mendling, J., Reijers, H.A. und van de Aalst, W.M.P. Seven Process Modeling Guidelines (7PMG). *Information and Software Technology*, 52 (2). 127-136 2010.
- [MeHS05] Mernik, M., Heering, J. und Sloane, A.M. When and how to develop domain-specific languages. *ACM Computing Surveys (CSUR)*, 37 (4) 2005.
- [MBKP04] Mertens, P., Bodendorf, F., König, W., Picot, A., Schuhmann, M. und Hess, T. *Grundzüge der Wirtschaftsinformatik*. Springer Verlag, Berlin, 2004.
- [MiCo11] Microsoft Corporation. *Microsoft Office Visio*. <http://office.microsoft.com/en-us/visio/>, Abruf am 2011-03-28.
- [Mill56] Miller, G. The marginal number seven, plus or minus two. *Psychological review* 1956, 81-97.
- [MiFr89] Mittal, S. und Fraymann, F. *Towards a Generic Framework of Configuration Task* 11th International Conference Artificial Intelligence (IJCAI '89), San Francisco, USA, 1989.
- [Mitt84] Mittelstraß, J. *Enzyklopädie Philosophie und Wissenschaftstheorie. Band 2, H-O*. Bibliographisches Institut AG, Mannheim, 1984.
- [Mitt95] Mittelstraß, J. *Enzyklopädie Philosophie und Wissenschaftstheorie. Band 3, P-So*. Verlag J.B. Metzler, 1995.
- [Mitt05] Mittelstraß, J. *Enzyklopädie Philosophie und Wissenschaftstheorie. Band 1, A-B*. Verlag J.B. Metzler, 2005.

- [Mobi00] Mobile Projektgruppe. *The Mobile Workflow Management Projekt*, <http://www6.informatik.uni-erlangen.de/research/projects/mobile/>, Abruf am 2010-11-23, 2000.
- [Mood05] Moody, D.L. Theoretical and practical issues in evaluating the quality of conceptual models: current state and future directions. *Data & Knowledge Engineering* 55. 243-276 2005.
- [Müll90] Müller, J. *Arbeitsmethoden der Technikwissenschaften*. Springer Verlag, Berlin, 1990.
- [MuRB08] Mutschler, B., Reichert, M. und Bumiller, J. Unleashing the effectiveness of process-oriented information systems: Problem analysis, critical success factors and implications *IEEE Transaction on Systems, Man and Cybernetics (Part C)*, 38. 208-291 2008.
- [NüRu02] Nüttgens, M. und Rump, J.F. Syntax und Semantik Ereignisgesteuerter Prozessketten (EPK). Desel, J. und Weske, M. eds. *Promise 2002 - Prozessorientierte Methoden und Werkzeuge für die Entwicklung von Informationssystemen, GI-Workshops und Fachgruppentreffens*, LNI, Potsdam, 2002, 64-77.
- [OMG06] Object Management Group (OMG). *Business Process Modeling Notation Specification, Version 1.0*. <http://www.omg.org/cgi-bin/doc?dtc/2006-02-01>. 2006
- [OMG11a] Object Management Group (OMG). *Business Process Model and Notation (BPMN) 2.0*. <http://www.omg.org/spec/BPMN/2.0/>, Abruf am 2011-05-25, 2011.
- [OMG11b] Object Management Group (OMG). *BPMN Implementors and Quotes*. http://www.bpmn.org/BPMN_Supporters.htm, Abruf am 2011-05-25, 2011.
- [OMG11c] Object Management Group (OMG). *Homepage*. <http://www.omg.org/>, Abruf am 2011-05-25, 2011.
- [OMG11d] Object Management Group (OMG). *Meta Object Facility (MOF)*. <http://www.omg.org/mof/>, Abruf am 2011-05-25, 2011.
- [OMG11e] Object Management Group (OMG) *Unified Modeling Language (UML)*. <http://www.uml.org/>, Abruf am 2011-05-25, 2011.
- [OoGC11] Office of Government Commerce. *IT Service Management (ITIL) Home Page*, <http://www.iti-officialsite.com/>, Abruf am 2011-05-25, 2011.
- [OnSo99] O'Neill, P. und Sohal, A.S. Business process reengineering: a review of recent literatur *Technovation*, 19 (9). 571-581 1999.
- [Opit04] Opitz, E. *Zur Notwendigkeit, Einführung und dauerhaften Nutzung klinischer Pfade*, <http://www.dvpflegewissenschaft.de/pdf/PfleGeOpitz0304.pdf>., 2004 Abruf am 2010-09-06

- [Orli72] Orlicky, J.A. Structuring the bill of materials for MRP. *Production and Inventory Management* (December). 19-42 1972.
- [Öste94] Österle, H. *Business Engineering – Prozess- und Systementwicklung*. Springer Verlag, Berlin, 1994.
- [OvAD06] Ouyang, C., van der Aalst, W.M.P., Dumas, M. und ter Hofstede, A. *From BPMN Process Models to BPEL Web Service* 4th International Conference on Web Services (ICWS), Chicago IL, USA, 2006.
- [Pahl70] Pahl, G. Wege zur Lösungsfindung. *Industrielle Organisation*, 39. 155-161 1970.
- [Pahl74] Pahl, G. Klären der Aufgabenstellung und Erarbeitung der Anforderungsliste *Konstruktion*, 24. 195-199 1974.
- [PaBe97] Pahl, G. und Beitz, W. *Konstruktionslehre - Methoden und Anwendung*. Springer Verlag, Berlin 1997.
- [Pete04] Peters, S. *Modell zur Beschreibung der kreativen Prozesse im Design unter Berücksichtigung der ingenieurtechnischen Semantik* Dissertation, Universität Duisburg-Essen, Duisburg-Essen, 2004.
- [Petr06] Petrov, I. *Meta-data, Meta-Modelling and Query Processing in Meta-data Repository Systems* Dissertation, Technische Fakultät der Universität Erlangen-Nürnberg, Erlangen, 2006.
- [Phal98] Phalp, K.T. The CAP-Framework for Business Process Modelling. *Information and Software Technology*, 40 (13). 731-744 1998.
- [PiRo95] Picot, A. und Rohrbach, P. Organisatorische Aspekte von Workflow-Management-Systemen. *Information Management* (1) 1995.
- [PBvL05] Pohl, K., Böckl, G. und van der Linden, F. *Software Product-line Engineering - Foundations, Principles and Techniques* Springer Verlag, Berlin, 2005.
- [Prod05] ProDatO Integration Technology GmbH. *Handbuch i>PM Integrated Process Manager*, Software-Dokumentation Erlangen 2005.
- [PSWW05] Puhlmann, F., Schnieders, A., Weiland, J. und Weske, M. PESOA - Variability Mechanism for Process Models *PESOA-Report No. TR 17/2005*, Hasso-Plattner-Institut, Potsdam, Germany, 2005.
- [RaWa09] Ralph, P. und Wand, Y. *A proposal for a formal definition of the design concept* Design Requirements Workshop 2009, Springer Verlag 2009.
- [RaJB04] Raumbaugh, J., Jacobson, I. und Booch, G. *The UML Reference Manual* Addison-Wesley, Massachusetts, USA, 2004.
- [Reck06] Recker, J. Process Modelling in the 21st Century. *BP Trends*, 3 (5). 1-6 2006.

- [Reck08] Recker, J. *Understanding Process Modeling Grammar Continuance - A Study Of The Consequences Of Representational Capabilities*, PhD, School of Information Systems at the Faculty of Information Technology Queensland University of Technology Brisbane, 2008.
- [Reck10] Recker, J. Opportunities and constraints: the current struggle with BPMN. *Business Process Management Journal*, 16 (1). 181-201 2010.
- [RIRG06] Recker, J., Indulska, M., Rosemann, M. und Green, P. *How good is BPMN really? Insight from Theory and Practice* 14th European Conference on Information Systems (ECIS'06), Goeteborg, Schweden, 2006.
- [RIRG09] Recker, J., Indulska, M., Rosemann, M. und Green, P. Business Process Modeling - A Comparative Analysis. *Journal of Association for Information Systems* 10 (4). 333-363 2009.
- [ReMA06] Recker, J., Mendling, J., Aalst, W.M.P. und Rosemann, M. *Model-driven Enterprise Systems Configuration* 18th International Conference on Advanced Information Systems Engineering (CAiSE'06), Luxembourg, 2006.
- [ReMe07] Recker, J. und Mendling, J. *Adequacy in Process Modeling: A Review of Measures and a Proposed Research Agenda* 8th Workshop on Business Process Modeling, Development and Support, Trontheim, 2007.
- [RRIG06] Recker, J., Rosemann, M., Indulska, M. und Green, P. *Business Process Modeling: A Maturing Discipline?* BPM Center Report BPM-06-20, BPMcenter.org, 2006.
- [ReRU00] Recker, J., Rosemann, M. und von Uthmann, C. *Guidlines of business process modeling* in van de Aalst, W.M.P., Desel, J. and Oberweis, A. eds. *Process Management. Models, Techniques and Empirical Studies*, Springer Verlag, Berlin, 2000, 30-49.
- [REFA85] REFA *Methodenlehre der Planung und Steuerung*. Hanser Fachbuch Verlag München, 1985.
- [ReRD05] Reichert, M., Rinderle, S. und Dadam, P. *Adaptive Process Management with ADEPT2* 21st International Conference on Data Engineering (ICDE'05), IEEE Computer Society Press Tokyo; Japan, 2005, 1113-1114.
- [Reij03] Reijers, H. *Design and Control of Workflow Processes: Business Process Management for the Service Industry*. Springer Verlag, 2003.
- [ReLi04] Reijers, H.A. und Liman, M.S. Best practice in business process redesign: an overview and qualitative evaluation of successful redesign heuristics *Omega* 2004.
- [RLvA03] Reijers, H.A., Liman, S. und van de Aalst, W.M.P. Product-Based Workflow Design *Journal of Management Information Systems* 20 (1). 229-262 2003.

- [ReMT09] Reijers, H.A., Mans, R.S. und Toorn, R.A. Improved model management with aggregated business models *Data & Knowledge Engineering*, 68 (2). 221-243 2009.
- [ReNV07] Reijers, H.A., Netjes, M. und Vanderfeesten, I. On pragmatic and formal process design approaches *8th Workshop on Business Process Modeling, Development, and Support*, Trontheim, 2007.
- [RevA05] Reijers, H.A. und van de Aalst, W.M.P. The Effectiveness of Workflow Management Systems: Predictions and Lessons Learned. *International Journal of Information Management*, 56 (5). 457-471 2005.
- [Remu02] Remus, U. Prozessorientiertes Wissensmanagement - Konzepte und Modellierung. Dissertation. Wirtschaftswissenschaftliche Fakultät, Universität Regensburg, Regensburg, 2002.
- [RiSt04] Richter-von Hagen, C. und Stucky, W. *Business-Process- und Workflow-Management: Prozessverbesserung durch Prozessmanagement*. B.G. Teubner Verlag, Wiesbaden, 2004.
- [Riit01] Riitahuhta, A. *Views and Experiences of Configuration Management*. in Riitahuhta, A. und Pulkinen, A. eds. Design for Configuration, Springer Verlag, Berlin, 2001.
- [Ritt00] Rittgen, P. Quo vadis EPK in ARIS? - Ansätze zu syntaktischen Erweiterungen und einer formalen Semantik. *Wirtschaftsinformatik*, 42 (1). 27-35 2000.
- [RoPr00] Rolland, C. und Prakash, N. From conceptual modelling to requirement engineering. *Annals of Software Engineering* (10). 151-176 2000.
- [RMTF99] Rosemann, B., Meerkamm, H., Trautner, S. und Feldmann, K. *Design for Recycling, Recycling Data Management and Optimal End-of-Life Planning based on Recycling Graphs* Int'l Conference on Engineering Design (ICED '99), Munich, Germany 1999.
- [Rose96] Rosemann, M. Multiperspektivische Informationsmodellierung auf der Basis der Grundsätze ordnungsmäßiger Modellierung. *Management & Computer* 4(4). 219-226 1996.
- [Rose06a] Rosemann, M. Potential Pitfalls of Process Modeling: part A. *Business Process Management Journal*, 12 (2). 249-254 2006.
- [Rose06b] Rosemann, M. Potential Pitfalls of Process Modeling: part B. *Business Process Management Journal*, 12 (3). 377-384 2006.
- [RoBr05] Rosemann, M. und de Bruin, T. *Application of a Holistic Model for Determining BPM Maturity* BP Trends, 2005.
- [RoSc97] Rosemann, M. und Schütte, R. *Grundsätze ordnungsmäßiger Referenzmodellierung*. Dr. Th. Gabler Verlag, Wiesbaden, Germany, 1997.

- [RovA07] Rosemann, M. und van der Aalst, W.M.P. A Configurable Reference Modeling Language. *Information Systems*, 32 (1). 1-23 2007.
- [Royc70] Royce, W.W. Managing the Development of Large Software Systems *IEEE WESCON*, The Institute of Electrical and Electronics Engineers, 1970.
- [RuLe95] Rubinovitz, J. und Levitin, G. Genetic algorithm for assembly line balancing. *International Journal of Production Economics*, 41. 343-354 1995.
- [RuBr95] Rummler, G.A. und Brache, A.P. *Improving Performance: How To Manage the White Space on the Organization Chart*. Jossey-Bass, San Fransisco, 1995.
- [RuRu94] Rupp, R.O. und Russell, J.R. The golden rules of process redesign. *Quality Progress*, 27 (12). 85-92 1994.
- [RvAH06] Russell, N., van der Aalst, W.M.P., ter Hofstede, A. und Wohed, P. On the Suitability of UML 2.0 Activity Diagramms for Business Process Modelling *BPM Center Report BPM-06-03*, BPMcenter.org, 2006.
- [SaWe97] Sabin, D. und Weigel, R. Product Configuration Frameworks-A Survey. *IEEE Intelligent Systems*, 13 (4). 42-49 1997.
- [SIBC07] Sadiq, S., Indulska, M., Bandara, W. und Chong, S. *Major Issues in Business Process Management: A Vendor Perspective* 11th Pacific Asia Conference on Information Systems (PACIS 2007) Auckland, New Zealand 2007.
- [Schä10] Schäling, B. Der moderne Softwareentwicklungsprozess mit UML <http://www.highscore.de/uml/>, 2010, Abruf am 2010-12-05.
- [SAKR05] Schäppi, B., Andreasen, M., Kirchgeorg, M. und Radermacher, F.J. *Handbuch Produktentwicklung*. Carl Hanser Verlag, München, Wien, 2005.
- [Sche94] Scheer, A.-W. *ARIS Toolset: A Software Product is Born*. Information Systems, 19 (8). 607-624 1994.
- [Sche99] Scheer, A.-W. *ARIS - Business Process Frameworks*. Springer Verlag, Berlin, 1999.
- [Sche00] Scheer, A.-W. *ARIS - Business Process Modeling*. Springer Verlag, Berlin, 2000.
- [Sche02] Scheer, A.-W. *ARIS – Vom Geschäftsprozess zum Anwendungssystem*. Springer Verlag, Berlin, 2002.
- [SBMS06] Scheer, A.-W., Boczanski, M., Muth, M., Schmitz, W.-G. und Segelbacher, U. *Bausteine und Prozesse im PLM*. In: Prozessorientiertes Product Lifecycle Management, Springer Verlag, Berlin 2006.
- [ScZi96] Scheer, A.-W. und Zimmermann, V. *Geschäftsprozessmanagement und integrierte Informationssysteme: Prozessmodellierung, Referenzmodelle und Softwaretechnologien*, Neuwied, 1996.

- [Schl04] Schlundt, M. *Historienverwaltung in Workflow-Management-Systemen* Dissertation, Technische Fakultät der Universität Erlangen-Nürnberg Erlangen, 2004.
- [ScSe08] Schmelzer, H.J. und Sesselmann, W. *Geschäftsprozessmanagement in der Praxis*. Carl Hanser Verlag, München, 2008.
- [Schn05] Schneider, B. *Design – Eine Einführung*. Birkhäuser Verlag 2005.
- [ScPu06] Schnieders, A. and Puhlmann, F. *Variability Mechanism in E-Business Process Families* 9th Int.'l. Conference on Business Information Systems (BIS'06), Klagenfurt, Austria, 2006.
- [Scho93] Schon, D.A. *The Reflective Practitioner: How Professionals Think in Action*, New York, 1993.
- [ScKl06] Schubert, K. und Klein, M. *Politiklexikon*. Dietz, Bonn, 2006.
- [Schu06] Schuh, G. *Change Management - Prozesse strategiekonform gestalten* Springer, Berlin, 2006.
- [Schü98] Schütte, R. *Grundsätze ordnungsmäßiger Referenzmodellierung: Konstruktion konfigurations- und anpassungsorientierter Modelle*. Gabler Verlag, Wiesbaden, 1998.
- [ScKr95] Schwarzer, B. und Krcmar, H. *Grundlagen der Prozessorientierung - eine vergleichende Untersuchung in der Elektronik- und Pharmaindustrie*. Deutscher Universitäts-Verlag, Wiesbaden, 1995.
- [Seid03] Seidewitz, E. What Models Means. *IEEE Software*. 26-32 2003.
- [Sieb08] Siebenhaar, K. *Unternehmen Universität. Wissenschaft und Wirtschaft im Dialog* VS Verlag für Sozialwissenschaften Wiesbaden, 2008.
- [Simo96] Simon, H.A. *The Sciences of the Artificial*, Cambridge, MA, 1996.
- [Sinu04] Sinur, J. *Magic Quadrant for Business Process Analysis*. Gartner Research Note Nr. M-22-0651 March, Gartner Inc., Stamford, Connecticut, 2004.
- [Sinz97] Sinz, E.J. *Analyse und Gestaltung universitärer Geschäftsprozesse und Anwendungssysteme* Informatik '97. Informatik als Innovationsmotor. 27. Jahrestagung der Gesellschaft für Informatik, Aachen, 1997.
- [SmDu01] Smith, H.S. und Duffy, A.H.B. *Product Structuring for Design Re-use*. In: Riitahuhta, A. und Pulkinen, A. eds. *Design for Configuration*, Springer Verlag, Berlin, 2001.
- [SmFi03] Smith, H. und Fingar, P. *Business Process Management - The Third Wave*. Meghan-Kiffer Press, Tampa, Florida, 2003.

- [SoRS07] Soffer, P., Reinhartz-Berger, I. und Sturm, A. *Facilitating Reuse by Specialization of Reference Models for Business Process Design* 19th Int'l. Conference on Advanced Information Systems Engineering (CAiSE'07) - Workshops and Doctoral Consortium Trondheim, Norway, 2007.
- [SoWa07] Soffer, P. und Wand, Y. Goal-Driven Multi-Process Analysis. *Journal of the Association for Information Systems*, 8 (3). 175-202 2007.
- [Soft11] Software AG. *ARIS Express*. <http://www.ariscommunity.com/aris-express>, Abruf am 2011-03-24
- [Stac73] Stachowiak, H. *Allgemeine Modeltheorie* Springer Verlag, Berlin 1973.
- [Stef00] Steffen, D. *Design als Produktsprache – Der „Offenbach Ansatz“ in Theorie und Praxis*. Verlag form Frankfurt / Main, 2000.
- [Ste99] Stein, K. *Integration von Anwendungsprozeßmodellierung und Workflow-Management*. Dissertation, Institut für Mathematische Maschinen und Datenverarbeitung, Technische Fakultät der Universität Erlangen-Nürnberg Erlangen, 1999.
- [StIv07] Stein, S. und Ivanov, K. EPK nach BPEL Transformation als Voraussetzung für praktische Umsetzung einer SOA. *Software Engineering*, 105. 75-80 2007.
- [Step01] Stephens, S. The Supply Chain Council and the Supply Chain Operations Reference Model. *Supply Chain Management - An International Journal*, 1 (1). 9-13 2001.
- [SvBo99] Svahnberg, M. and Bosch, J. *Issues Concerning Variability in Software Product Lines* Int'l Workshop on Software Architectures for Product Families (IW-SAPF-3), Las Palmas de Gran Canaria, Spain, 1999.
- [TaVJ10] Talib, R., Volz, B. and Jablonski, S. *Agent Assignment for Process Management: Goal Modeling for Continuous Ressource Management* 6th. Int'l. Workshop on Business Process Design (BPD 2010), Hoboken, New Jersey, USA 2010.
- [TeUI09] Technischer Universität Ilmenau *Qualitätsmanagementhandbuch*, Ilmenau, 2009.
- [tHAA10] ter Hofstede, A., van der Aalst, W.M.P., Adams, M. und Rusell, N. *Modern Business Process Automation - YAWL and its Support Environment*. Springer Verlag, Berlin, 2010.
- [Tesc03] Teschke, T. *Semantische Komponentensuche auf der Basis von Geschäftsprozessmodellen*, Dissertation, Carl von Ossietzky Universität Oldenburg, Oldenburg, 2003.
- [Teub99] Teubner, R.A. *Organisations- und Informationssystemgestaltung, theoretische Grundlagen und integrierte Methoden*, Dissertation Westfälische Wilhelms-Universität Münster, Münster, 1999.

- [ThEF11a] The Eclipse Foundation. *Eclipse Modeling Framework Project (EMF)*. <http://www.eclipse.org/modeling/emf/>, Abruf am 2011-05-26, 2011
- [ThEF11b] The Eclipse Foundation. *Graphical Modeling Project (GMP)*. <http://www.eclipse.org/modeling/gmp/>, Abruf am 2011-05-26, 2011
- [ThEF11c] The Eclipse Foundation. *Xtext*. <http://www.eclipse.org/Xtext/>, Abruf am 2011-05-26, 2011
- [Thom05] Thomas, O. *Understanding the Term Reference Model in Information Systems Research: History, Literature Analysis and Explanation* Workshop on Business Process Reference Models (BPRM'05) Nancy, France, 2005.
- [TiSo97] Tiuhonen, J. und Soinen, T. *Product Configurators - Information System Support for Configurable Products Increasing Sales Productivity through the Use of Information Technology during the Sales Visit. A Survey of the European Market*, Hewson Consulting Group, 1997.
- [ToKe04] Tolvanen, J.-H. und Kelly, S. Domänenspezifische Modellierung. *OBJECT spectrum* (4). 30-35 2004.
- [TrGC95] Tryggeseth, E., Gulla, B. und Conradi, R. *Modelling Systems with Variability using the PROTEUS Configuration Language* Int'l Conference on Software Configuration Management (ICSE'95) and SCM-5 Workshops, 1995.
- [TsJu95] Tsalgatidou, A. und Junginger, S. *Modelling in the Re-Engineering Process*. ACM SIGOIS Bulletin, 16 (1). 17-24 1995.
- [Tsic97] Tsichritsis, D. The Dynamics of Innovation. In *Beyond Calculation: The Next Fifty Years of Computing*, Copernicus, 1997, 259-265.
- [UnBa10] Universität Bayreuth: *Handbuch „Prozessqualität für Studium und Lehre der Universität Bayreuth“*, Universität Bayreuth, Bayreuth, 2010.
- [UnBa11] Universität Bayreuth: Qualitätsmanagement. <http://www.uni-bayreuth.de/universitaet/qualitaetsmanagement/index.html>. Abruf am: 2011-03-20
- [vAal98] van der Aalst, W.M.P. The application of Petri Net to Workflow Management *The Journal of Curcuits, Systems and Computers*, 8 (1). 21-66 1998.
- [vAal99] van der Aalst, W.M.P. Formalization and Verification of Event-driven Process Chains. *Information and Software Technology*, 41 (10). 639-650 1999.
- [vADR05] van der Aalst , W.M.P., Dreiling, A., Rosemann, M. und Jansen-Vullers, M.H. *Configurable Process Models as a Basis for Reference Modeling* Business Process Management Workshops (BPM'05), Nancy, France, 2005.

- [vADG10] van der Aalst, W.M.P., Dumas, M., Gottschalk, F., ter Hofstede, A., La Rosa, M. und Mendling, J. Preserving Correctness During Business Process Model Configuration. *Formal Aspects of Computing*, 22 (3). 459-482 2010.
- [vAJa00] van der Aalst, W.M.P. and Jablonski, S. Dealing with workflow change: identification of issues and solutions *International Journal of Computer Systems Science & Engineering* 15 (5). 267-276 2000.
- [vAtH05] van der Aalst, W.M.P. und ter Hofstede, A. YAWL: Yet Another Workflow Language. *Information Systems*, 30 (4). 245-275 2005.
- [vAHW03] van der Aalst, W., Hofstede ter, A. und Weske, M. *Business Process Management: A Survey* in 1st International Conference on Business Process Management (BPM 2003), Springer Verlag, Berlin, 2003.
- [vDKV00] van Deursen, A., Klint, P. und Visser, J. Domain Specific Languages: An Annotated Bibliographie. *ACM SIGPLAN Notices*, 35 (6) 2000.
- [vHRe00] van Hee, K.M. und Reijers, H.A. *Using Formal Analysis Techniques in Business Process Redesign* Business Process Management (BPM 2000), Springer Verlag, 2000.
- [VDA05] VDA Verband der Automobilindustrie. *VDA Rekommandation 4965 T1. Engineering Change Management (ECM) - Part 1: Engineering Change Request (ECR) Version 1.1*, 2005.
- [VDI82] VDI Verein Deutscher Ingenieure. *Richtlinie 2222 Blatt 2: Konstruktionsmethodik; Erstellung und Anwendung von Konstruktionskatalogen*, VDI Verlag, Düsseldorf, 1982.
- [VDI93] VDI Verein Deutscher Ingenieure. *Richtlinie 2221: Methodik zum Entwickeln und Konstruieren technischer Systeme und Produkte*, VDI-Verlag, Düsseldorf, 1993.
- [VDI97] VDI Verein Deutscher Ingenieure. *Richtlinie 2222 Blatt 1: Konstruktionsmethodik - Methodisches Entwickeln von Lösungsprinzipien*, VDI Verlag, Düsseldorf, 1997.
- [VDI04] VDI Verein Deutscher Ingenieure. *Richtlinie 2223: Methodisches Entwerfen technischer Produkte*, VDI Verlag, Düsseldorf, 2004.
- [VDRE76] VDI/REFA. *Elektronische Datenverarbeitung bei der Produktionsplanung und -steuerung* VDI, Düsseldorf 1976.
- [VTMR07] Vergidis, K., Tiwari, A., Majeed, B. und Roy, R. Optimisation of Business Process Designs: An algorithmic approach with multiple objectives. *International Journal of Production Economics*, 109 (1-2). 105-121 2007.
- [VeCo94] Vessey, I., und Conger, S. A. (1994). Requirement Specification: Learning Object, Process, and Data Methodologies. *Communications of the ACM*, 37(5), 102-113.

- [VoZe09] Voigtmann, P. und Zeller, T. *Fachvortrag - Beiträge zur Integrationsproblematik im Kontext von Electronic Business und Elektronischen Marktplätzen* 6. Internationale Tagung der Wirtschaftsinformatik - Medien - Märkte - Mobilität Dresden, 2009.
- [Volz11] Volz, B. *Werkzeugunterstützung für methodenneutrale Metamodellierung* Dissertation, Fakultät für Mathematik, Physik und Informatik, Universität Bayreuth, Bayreuth, 2011.
- [VoJa10a] Volz, B. und Jablonski, S. *OMME - A Flexible Modeling Environment* Workshop on Flexible Modeling Tools (FlexiTools@SPLASH 2010), Reno/Tahoe, Nevada, USA, 2010.
- [VoJa10b] Volz, B. und Jablonski, S. *Towards an Open Meta Modeling Environment* 10th Workshop on Domain-Specific Modeling (DSM 2010) Reno/Tahoe, Nevada, USA, 2010.
- [vBro03] vom Brocke, J. Referenzmodellierung. Gestaltung und Verteilung von Konstruktionsprozessen (Dissertation) in Becker, J., Grob, H.L., Klein, M., Kuchen, H., Müller-Funk, U. und Vossen, G. eds. *Advances in Information Systems and Management Science* Logos Verlag, Berlin, 2003.
- [vBRo10] vom Brocke, J. und Rosemann, M. *Handbook on Business Process Management: Strategic Alignment, Governance, People and Culture*. Springer Verlag, Berlin, 2010.
- [WaKü08] Wahler, K. und Küster, J.M. *Predicting coupling of object-centric business process implementations* 6th Int.'l. Conference on Business Process Management (BPM'08), Milan, Italy, 2008.
- [WaWE92] Walls, J.G., Widmeyer, G.R. und El Sawy, O.A. Building an Information Systems Design Theory for Vigilant EIS. *Information Systems Research*, 3 (1). 36-59 1992.
- [WICS07] Wasana, B., Indulska, M., Chong, S. und Sadiq, S. *Major Issues in Process Management: An Expert Perspective*. 15th European Conference on Information Systems (ECIS'07), St. Gallen, Switzerland, 2007.
- [Wats08] Watson, A. *UML® vs. DSLs: A false dichotomy*, Object Management Group, 2008.
- [WeRe08] Weber, B. und Reichert, M. *Refactoring Process Models in Large Process Repositories* 20th international conference on Advanced Information Systems Engineering (CAiSE'08), Montpellier, France, 2008.
- [Wede94] Wedekind, H. Are the Terms "Version" and "Variant" Orthogonal to One Another? - A Critical Assessement of the STEP Standardization. *SIGMOD*, 23 (4). 3-7 1994.
- [WeMü81] Wedekind, H. und Müller, T. Stücklistenorganisation bei einer großen Variantenzahl. *Angewandte Informatik*, 9. 377-383 1981.

- [Weis91] Weis, E. (ed.) *Pons-Kompaktwörterbuch - Englisch-Deutsch, Deutsch-Englisch*. Klett Verlag, Stuttgart, 1991.
- [Wesk07] Weske, M. *Business Process Management - Concepts, Languages, Architectures* Springer, Berlin, Heidelberg, New York, 2007.
- [Whit04] White, S. *Introduction to BPMN*, <http://www.bpmn.org/>, Abruf am 2010-11-26, 2004.
- [Wild07] Wildemann, H. *Projektmanagement - Leitfaden zu Koordination und Controlling von Funktionsübergreifenden Projekten* TCW-Verlag, München, 2007.
- [Wilh07] Wilhelm, R. *Prozessorganisation*. Oldenbourg Wissenschaftsverlag GmbH, München, 2007
- [Wino96] Winograd, T. *Bringing Design to Software*. Addison-Wesley, Inc., Reading, MA, 1996.
- [Wino97] Winograd, T. The Design of Interaction. In Denning, P. und Metcalfe, R. eds. *Beyond Calculation, The Next 50 Years of Computing*, Springer-Verlag, New York, 1997, 149-162.
- [WKWI08] Wissenschaftliche Kommission Wirtschaftsinformatik im Verband der Hochschullehrer für Betriebswirtschaft e. V. (WKWI) und Fachbereich Wirtschaftsinformatik der Gesellschaft für Informatik (GI-FB WI). WI - Orientierungsliste. *Wirtschaftsinformatik*, 50 (2). 155-163 2008.
- [Witz00] Witzl, A. *Das problemzentrierte Interview*. Forum Qualitative Sozialforschung, Volume 1 (1) 2000.
- [WoHa10] Wolf, C. und Harmon, P. *The State of Business Process Management Business Process Trends* BPTrends Report. 2010.
- [WoMC11] Workflow Management Coalition (WfMC). *XPDL*. <http://www.wfmc.org/xpdl.html>, Abruf am 2010-11-22, 2011
- [YuLi00] Yu, C.-S. und Li, H.-L. A robust optimization model for stochastic logistic problems. *International Journal of Production Economics*, 64. 385-397 2000.
- [Zair94] Zairi, M. *Practical Benchmarking - A Complete Guide*. Chapman & Hall, London, 1994.
- [ZaSi95] Zairi, M. und Sinclair, D. Business process re-engineering and process management - A survey of current practice and future trends in integrated management. *Business Process Re-engineering & Management Journal*, 1 (1). 8-30 1995.

- [ZdKa02] Zdrowomyslaw, N. und Kasch, R. *Betriebvergleiche und Benchmarking für die Managementpraxis: Unternehmensanalyse, Unternehmenstransparenz und Motivation durch Kenn- und Vergleichsgrößen*. Oldenbourg Verlag, München, 2002.
- [zMRo04] zur Mühlen, M. und Rosemann, M. *Multi-Paradigm Process Management*. 5th Workshop on Business Process Modeling, Development and Support (BPMDS 2004), Riga, Latvia, 2004.

Anhang A

Leitfaden - Interview

Einleitung

Wie wird Prozessmanagement umgesetzt?

Erst einmal vielen Dank, dass Sie sich die Zeit für unser Gespräch genommen haben.

Ich habe ja am Telefon/gegebenenfalls Infoblatt schon kurz umrissen, worum es in unserem Gespräch gehen soll.

Mich interessiert, wie Sie dieses Thema bewerten und einschätzen. Wir befragen Unternehmen aus verschiedenen Branchen, unter anderem auch Ihre.

Vorab möchte ich Sie schon einmal darauf hinweisen, dass die Fragen, die ich Ihnen im Verlauf des Gesprächs stellen möchte, nicht unbedingt meinen eigenen Ansichten entsprechen.

Bei unserem Gespräch gibt es keine richtigen oder falschen Antworten. Ich möchte Sie bitten, mir Ihre ganz persönliche Meinung und Ihre Erfahrungen zu erzählen.

Ihre gesamten Angaben werden anonym und vertraulich behandelt. Wenn bei Ihnen während des Interviews Fragen auftauchen, können Sie diese natürlich jederzeit stellen.

Bevor wir jetzt speziell auf das Thema „Umsetzung des Prozessmanagements“ kommen, würde mich zunächst einmal ganz allgemein interessieren ...

Interviewleitfaden – Themenübersicht

I. Einführung

II. Rahmenbedingungen

III. Prozesse (Erfassung und Verwendung)

IV. Informationstechnologie

V. Mitarbeiter

[Bemerkung: die unterstrichenen Fragen sollen auf jeden Fall gestellt werden; die Kursiv gedruckten Bereiche dienen dazu, dem Interviewten bei der Strukturierung seiner Antworten behilflich zu sein]

I. Einführung

... ist Prozessmanagement oder Prozessorientierung bei Ihnen ein Thema?

II. Rahmenbedingungen

Das Unternehmen und seine Struktur bilden das grundlegende Gerüst, den Rahmen zur Durchführung der einzelnen Aufgaben im Geschäftsalltag und beeinflussen daher wesentlich das Prozessmanagement. Uns interessiert daher: ...

A. Strategische Ebene

1. Wie stellt sich die Aufbauorganisation des Unternehmens, d.h. die Strukturierung der organisatorischen Einheiten dar?

... bzgl. Hierarchieebenen (viele, wenige)?

... bzgl. Geschäftseinheiten, - Bereichen, Abteilungen?

...bzgl. Einlinien,- Mehrlinienorganisation/ funktionale, divisionale, Matrixorganisation?

2. Ist ihr Unternehmen zertifiziert z.B. nach ISO 9001?

... Ja: Spielt PM dabei eine Rolle

(a) Ja: Inwiefern?

(b) Nein: →

... Nein: →

3. Inwiefern verwendet das Management Prozesse zur Planung/ Steuerung/ Kontrolle/ etc.?

B. Operative Ebene

1. Wie erfolgt eine Strukturierung der Abläufe im Unternehmen?

... als Prozessorganisation?

... welche anderen Konzepte gibt es?

2. Sind (Standard-) Vorgehensmodelle bei Ihnen ein Thema?

... VDI 2221/ RUP/ V-Modell

III. Prozesserfassung und -Verwendung

Dreh- und Angelpunkt beim Prozessmanagement sind die Prozesse. Zur Umsetzung des Prozessmanagements sollten die Prozesse jedoch entsprechend aufbereitet werden.

A. Identifizierung/Modellierung

Werden Abläufe bei Ihnen erfasst bzw. dokumentiert?

Und zwar nicht während der Aus-/ Durchführung der Prozesse, sondern es soll dabei um die Identifizierung/ das Erkennen der Prozesse gehen.

... wenn ja: →

(1) Werden Prozesse in allen Bereichen/ Ebenen des Unternehmens erfasst?

...ja/ nein: → Warum?

(2) Wie werden die Prozesse erfasst bzw. identifiziert werden?

(3) Wo liegt der inhaltliche Schwerpunkt (der Erfassung bzw. Darstellung)?

... d.h. welche Aspekte werden betrachtet bzw. fokussiert?

... welche Aspekte fehlen ihrer Meinung nach?

(4) Welche Anforderungen werden an die Erfassung bzw. Darstellung gestellt?

... fehlerfrei, vollständig, subjektiv, informeller Charakter.... (inhaltlich)

... syntaktisch korrekt, ausgezeichnete Ausdrücke, berechenbar ... (formal)

...wenn nein: →

(1) Aus welchen Gründen werden bei Ihnen Prozesse nicht identifiziert?

... z.B. wird es nicht als notwendig erachtet

... z.B. wurde daran bisher nicht gedacht

B. Verwendung/ Umsetzung:

1. Wie werden Prozesse bei Ihnen umgesetzt/ eingesetzt/ verwendet?

...Darstellung/ Illustration der Abläufe im Unternehmen (zur Kommunikation/ Training,...)?

... Ausführung/ Umsetzung durch ein technisches System (mit /ohne Interaktion des Menschen)?

... Monitoring von Vorgängen?

... Optimierung von Vorgängen?

2. Helfen Ihnen Prozesse ihre Arbeit zu strukturieren?

IV. Informationstechnologie

Insbesondere die Informationstechnologie hat Einfluss auf das Prozessmanagement. Sie beeinflusst die operativen Abläufe im Unternehmen. Uns interessiert daher:

A. Aufbau/ Konzept

1. Verwenden Sie ein globales System oder arbeiten Sie mit verschiedenen Anwendungen?

... verschiedene Anwendungen: →

(1) Besteht eine Integration zwischen den verschiedenen Anwendungen?

... über Prozesse?

... über eine zentrale Datenbank, ...?

... globales System: →

(2) Wonach ist das globale System strukturiert/ gegliedert?

... nach Funktionen?

... nach Prozessen?

B. Funktionalität

1. Welche Arten von Anwendungen/Technologien/Informationstechnologien werden verwendet?

(Umformuliert = Wie ist die Ausrichtung ihrer Technologien?)

... Datenverwaltung, z.B. als Kundenverwaltungssystem?

... Abrechnung, z.B. Reisekostenabrechnung?

... Planungssystem/ ERP-Systeme, z.B. zur Ressourcenplanung, Materialwirtschaft?

2. Wie erfolgt die Informationsversorgung des Anwenders?

... über eine zentrale/ dezentrale Speicherung?

... durch automatisches Weiterleiten, persönliches Weiterleiten der Daten?

3. Gibt es einen Zusammen/ eine Kopplung/ eine Verbindung zwischen den Prozessen und der IT?

C. Operative Ebene/ Umsetzung/ Verwendung

1. Wie finden Anwender für ihre Tätigkeiten geeignete Anwendungen bzw. Informationssysteme, d.h. wie/womit werden die Funktionen dem Anwender vermittelt?

... Systeme leiten Anwender komplett...

... Prozessbeschreibungen ...

2. Welche Technologien fehlen in Ihren Betrieb/ ihre Abläufe und sollten daher als nächstes erworben oder auch entwickelt werden?

V. Mitarbeiter

Zur Umsetzung des Prozessmanagements tragen im Wesentlichen auch die Mitarbeiter bei da durch sie das Prozessmanagement quasi im Alltag gelebt wird. Daher interessieren uns zum einen der Berufsalltag im Unternehmen, wie auch einige Aspekte zur Qualifikation/ Fortbildung der Mitarbeiter.

A. Berufsalltag:

1. Wie werden (neuen) Mitarbeitern die Abläufe im Unternehmen vermittelt?

...Coaching, Training...?

... Prozessmodell/ Prozesshandbücher?

2. Wie ist ein Mitarbeiter in den Geschäftsablauf eingebunden?

... über Prozesse/ Netzwerk bzw. Prozessgeflecht?

...als isoliertes Teilgebiet?

3. Wo besteht Optimierungsbedarf bzgl. eines effektiveren Arbeitsablaufs?
























B. Qualifikation/ Fortbildung:

1. Inwieweit wird Wert auf Kenntnisse über Prozessmanagement gelegt?















2. Werden die Mitarbeiter in Prozessen geschult?

Anhang B

BPMN Events

	Start			Zwischen				Ende
	Top-Level	Ereignis-Teilprozess unterbrechend	Ereignis-Teilprozess Nicht-unterbrechend	Eingetreten	Angehaftet unterbrechend	Angehaftet nicht-unterbrechend	Ausgelöst	
Blanko: Untypisierte Ereignisse, i. d. R. am Start oder Ende eines Prozesses								
Nachricht: Empfang und Versand von Nachrichten								
Timer: Periodische zeitliche Ereignisse, Zeitpunkte oder Zeitspannen								
Eskalation: Meldung an den nächst höheren Verantwortlichen								

Bedingung: Reaktion auf veränderte Bedingungen und Bezug auf Geschäftsregeln								
Link: Zwei zusammengehöri- ge Link- Ereignisse repräsentieren einen Sequenzfluss								
Fehler: Auslösen und behandeln von definierten Fehlern								
Abbruch: Reaktion auf abgebrochene Transaktionen oder Auslösen von Abbrüchen								
Kompensation: Behandeln oder Auslösen einer Kompensation								
Signal: Signal über mehrere Prozesse. Auf ein Signal kann mehrfach reagiert werden								

Mehrfach: Eintreten eines von mehreren Ereignissen. Auslösen aller Ereignisse								
Mehrfach/ Parallel: Eintreten aller Ereignisse								
Terminierung: Löst die sofortige Beendigung des Prozesses aus								